C O N T E N T S

1. INTRODUCTION TO OrCAD/SDT

   1.1. Features of OrCAD/STD
        1.1.1. DRAFT
        1.1.2. Part Libraries
        1.1.3. DECOMP
        1.1.4. COMPOSER
        1.1.5. Other Utility Programs
   1.2. How to Use This Manual
   1.3. Troubleshooting
        1.3.1. If You Have Questions

2. SETTING UP OrCAD/STD

   2.1. What You Need to Begin
   2.2. Preparing for Inatallation
        2.2.1. Making Backup Copies
   2.3. Installing OrCAD/STD
        2.3.1. Installing OrCAD/STD on a Dual Floppy System
        2.3.2. Installing OrCAD/STD on a Hard-Disk System
        2.3.3. Installing a Mouse
   2.4. Configuring OrCAD/STD
        2.4.1. Invoking the OrCAD/STD Configuration File
        2.4.2. Driver Prefix
        2.4.3. Display Driver
        2.4.4. Printer Driver
        2.4.5. Plotter Driver
        2.4.6. Library Prefix
        2.4.7. Libray File
        2.4.8. Worksheet Prefix
        2.4.9. Macro File
        2.4.10. Initial Macro
        2.4.11. Macro Buffer Size
        2.4.12. Hierarchy Buffer Size
        2.4.13. Color Table/Plotter Pen Table
        2.4.14. Update Configuration Information
        2.4.15. Returning to DOS with Quit
        2.4.16. Run Program
        2.4.17. Dual-Floppy Configuration Example
        2.4.18. Sample Hard Disk Configuration
   2.5. If You Have Configuration Problems
   2.6. Send in Your Registration Card

3. GETTING STARTED

   3.1. Invoking DRAFT
   3.2. Command and Menu Organization
        3.2.1 Invoking Commands
   3.3. Creating a New Worksheet
   3.4. Loading Worksheet Files
        3.4.1. From DOS
        3.4.2. After Invoking DAFT
        3.4.3. From Within DRAFT
   3.5. Saving Worksheets to a File
   3.6. Updating Files
   3.7. Exiting DRAFT

3.8. Printing a File
3.9. Plotting File
3.10. Using a Mouse
3.11. Worksheet File Structures
3.11.1. The Flat File Structure
3.11.2. The Hierarchy File Structure
3.11.3. The One-Sheet File Structure

4. COMMANDS

4.1. AGAIN
4.2. BLOCK
4.2.1. BLOCK Move
4.2.2. BLOCK Drag
4.2.3. BLOCK Fixup
4.2.4. BLOCK Save
4.2.5. BLOCK Get
4.2.6. BLOCK Import
4.2.7. BLOCK Export
4.3. CONDITIONS
4.3.1. Worksheet Memoy Size
4.3.2. Free Hierarchy Buffer
4.3.3. Free Macro Buffer
4.3.4. Free System Memory
4.4. DELETE
4.4.1. DELETE Object
4.4.2. DELETE Block
4.4.3. DELETE Undo
4.5. EDIT
4.5.1. Editing Labels
4.5.2. Editing Power Objects
4.5.3. Editing Sheets
4.5.4. Editing Parts
4.5.5. Editing the Title Block
4.6. FIND
4.7. GET
4.7.1. The Outline Symbol
4.7.2. Rotating and Placing Parts
4.8. HARDCOPY
4.8.1. HARDCOPY Destination
4.8.2. HARDCOPY File Mode
4.8.3. HARDCOPY Make Hardcopy
4.8.4. HARDCOPY Print Mode
4.8.5. HARDCOPY Width of Paper
4.9. JUMP
4.9.1. JUMP Tag
4.9.2. JUMP Reference
4.9.3. JUMP X Location
4.9.4. JUMP Y Location
4.10. LIBRARY
4.10.1. LIBRARY Directory
4.10.2. LIBRARY Browse
4.11. MACRO
4.11.1. Valid Macro Key Names
4.11.2. MACRO Capture
4.11.3. MACRO Examples
4.11.4. Terminating a Macro Between Commands
4.11.5. Capturing Macro and Pausing
4.11.6. Executing a Macro
4.11.7. MACRO Delete

4.11.8. MACRO Initialize
4.11.9. MACRO List
4.11.10. MACRO Write
4.11.11. MACRO Read
4.11.12. MACRO File Format
4.11.13. Nesting Macros
4.12. PLACE
4.12.1. PLACE Wire
4.12.2. PLACE Bus
4.12.3. PLACE Junction
4.12.4. PLACE Entry (Bus)
4.12.5. PLACE Label
4.12.6. Correct Label Positioning
4.12.7. PLACE Module Port
4.12.8. PLACE Power
4.12.9. PLACE Sheet
4.13. QUIT
4.13.1. QUIT Enter Sheet
4.13.2. QUIT Leave Sheet
4.13.3. QUOT Update File
4.13.4. QUIT Write File
4.13.5. QUIT Initialize
4.13.6. QUIT Suspend to DOS
4.13.7. QUIT Abandon Edits
4.14. REPEAT
4.15. SET
4.15.1. SET Auto Pen
4.15.2. SET Backup File
4.15.3. SET Drag Buses
4.15.4. SET Error Bell
4.15.5. SET Left Button
4.15.6. SET Macro Prompts
4.15.7. SET Orthogonal
4.15.8. SET Show Pin Numbers
4.15.9. SET Title Block
4.15.10. SET Worksheet Size
4.15.11. SET X, Y Display
4.15.12. SET Grid Parameters
4.15.13. SET Repeat Parameters
4.16. TAG
4.17. ZOOM
4.17.1. ZOOM Center
4.17.2. ZOOM In
4.17.3. ZOOM Out
4.17.4. ZOOM Select

5. HIERRCHY

5.1. Elements of a Hierarchy
5.2. The Place Sheet Command
5.3. Types of Hierarchical Designs
5.4. Example of a Simple Hierarchy
5.4.1. Invoking ANNOTATE on CMOSCPU.SCH
5.4.2. Invoking ERCHECK on CMOSCPU.SCH
5.4.3. Invoking TREELIST on CMOSCPU.SCH
5.4.4. Invoking NETLIST on CMOSCPU.SHC
5.4.5. Invoking PARTLIST on CMOSCPU.SCH
5.3. Complex Hierarchies
5.6. An Example of a Complex Hierarchy
5.6.1. Power Objects

5.6.2. Further Complex Hierachy Example Discussion
5.6.3. Creating the Annotation File
5.6.4. Invoking the ERCHECK Utility
5.6.5. Invoking the TREELIST Utility
5.6.6. Invoking the NETLIST Utility
5.6.7. Invoking the PARTLIST Utility
5.6.8. Invoking the PRINTALL Utility

6. UTILITIES

6.1. Utility Program Format Notation
6.2. Invoking Utility Programs
6.2.1. On Hierarchy File Structures
6.2.2. On Flat File Structures
6.2.3. On One Sheet File Structures
6.2.4. Two Floppy Disk Applications
6.3. ANNOTATE
6.3.1. Invoction Examples Using Hierarchical Structured Files
6.3.2. Invocation Example Using Flat File Structure
6.3.3. On Sheet File Structure Invocation Example
6.3.4. The Annotation File
6.4. BACKANNO
6.4.1. Invocation Example Using a Hierarchical Structured File
6.4.2. Invocation Example Using a Flat File Structure
6.4.3. Invocation Example Using a One Sheet File Structure
6.4.4. Back Annotating Schematics Based on an Annotation File
6.4.5. WAS/IS File Format
6.5. CLEANUP
6.5.1. Invocation Example Using Hierarchical Structured Files
6.5.2. Invocation Example Using Flat File Structures
6.5.3. Invocation Example Using a One Sheet File Structure
6.5.4. Checking Schematics Based on an Annotation File
6.6. ERRCHECK
6.6.1. Invocation Examples Using Hierarchical Sructured Files
6.6.2. Invocation Examples Using Flat File Structures
6.6.3. Invocation Example Using a One Sheet File Strucutre
6.6.4. Checking Schematics Based on an Annotation File
6.6.5. ERCHECK Messages and Resolutions
6.7. NETLIST
6.7.1. Invocation Examples Using Hierachical Structured Files
6.7.2. Invocation Examples Using Flat File Structures
6.7.3. Invocation Example Using a One Sheet File Structure
6.7.4. Net Lists Based on an Annotation File
6.7.5. The Limits of Netlist
6.7.6. Notes on Particular Formats
6.7.7. Applications of Internal Labels
6.7.8. Applications of Bus Member Labels
6.7.9. Applications of Module Ports

6.7.10. Splitting Buses
6.7.11. Multiple Labels on a Bus
6.7.12. Combining Internal and Bus Member Labels
6.7.13. Connecting Bus Labels to Modul Ports
6.7.14. Processing Module Ports and Internal Labels
6.7.15. Handling Power
6.7.16. Connecting Different Power Object Nemes
6.7.17. Handling and Isolating Grounds
6.7.18. Connecting Power Objects to a Module Port
6.7.19. Considertions when Handling Power in a Hierarchy
6.7.20. Example of Isolating Power-Battery Backup
6.7.21. Handling Physical Connectors
6.7.22. CONFIG.SYS File
6.8. PARTLIST
6.8.1. Invocation Examples Using Hierarchical Structured Files
6.8.2. Invocation Examples Using Flat Files Structures
6.8.3. Invocation Example Using a One Sheet File Structure
6.8.4. Obtaining a Parts List from Schematics Based on an Annotation File
6.8.5. Including File Format
6.8.6. Sample PARTLIST Output
6.8.7. Sample PARTLIST Output Using an Include File
6.9. PLOTALL
6.9.1. Invocation Examples Using Hierarchicl Structured Files
6.9.2. Invocation Examples Using Flat File Structures
6.9.3. Invocation Example Using a One Sheet File Structure
6.9.4. Invocation Example - Directing the Output to a File
6.9.5. Plotting Schematics Based on Annotation Files
6.9.6. Setting up the Plotter Configuration
6.9.7. Plotter Cable Wiring Diagrams
6.9.8. Plotter Problems
6.9.9. Output Scaling
6.9.10. Plotter Hints
6.9.11. HP Plotters
6.9.12. HI Plotters
6.9.13. Suppressing the Title Block and Border
6.10.PRINTALL
6.10.1. Invocation Examples Using Hierarchical Structured Files
6.10.2. Invocation Examples Using Flat File Structures
6.10.3. Invocation Examples Using a One Sheet File Structure
6.10.4. Printing Schemtics Based on Annotation Files
6.11.TREELIST
6.11.1. Invocation Examples Using Hierarchical Structured Files
6.11.2. Sample Output from the TREELIST Utility

7. LIBRARIES

7.1. Library Development Process Overview
7.2. The COMPOSER and DECOMP Utilities
7.2.1. Invoking COMPOSER
7.2.2. Invoking DECOMP
7.3. Creating a Source File
7.3.1. The Prefix Definition
7.3.2. The Part Definition
7.3.3. Block Symbol Definition
7.3.4. Bitmap Symbol Definition
7.4. SDL Reference
7.4.1. Syntax Diagram
7.4.2. Textual Representation of Syntax
7.4.3. Prefix Definition
7.4.4. Part Definition
7.4.5. Pin Definition
7.4.6. Bitmap Definition
7.4.7. Conversion Bitmap
7.5. A Final Example: Making a Connector
7.6. Examples of Source Libraries
7.6.1. SAMPLE 1.STC: A block Symbol Library Source File
7.6.2. SAMPLE 2.SRC: A TTL Block Symbol Library Source File
7.6.3. SAMPLE 3.SRC: A Bibmap Symbol Library Source File

TUTORIAL

8.1. Assumption
8.2. Requirements
8.3. Module 1: Constructing a Worksheet, Part 1
8.3.1. Invoking DRAFT
8.3.2. Setting the DRAFT Environment
8.3.3. Using the Libraries
8.3.4. Connecting Wires
8.3.5. Making Junctions
8.3.6. Editing the Title Block
8.3.7. Updating the Worksheet
8.3.8. Making a Hardcopy
8.3.9. Exiting DRAFT
8.4. Module 2: Constructing a Worksheet, Part 2
8.4.1. Creating the Custom Library
8.4.2. Invoking DRAFT and Setting macro
8.4.3. Placing Module Ports
8.4.4. Placing a Bus
8.4.5. Placing Power Objects
8.4.6. A Complete Circuit
8.4.7. Placing More Parts
8.4.8. Editing the Worksheet
8.5. Module 3: Using the OrCAD Utilities
8.5.1. ANNOTATE
8.5.2. BACKANNO
8.5.3. CLEANUP
8.5.4. ERCHECK
8.5.5. NETLIST
8.5.6. PARTLIST
8.5.7. PRINTALL, PLOTALL
8.6. Conclusion

APPENDIX

A Summary of Library Components
B Driver Configurations

.pn8

1. INTRODUCTION TO OrCAD/SDT

OrCAD Schematic Design Tools (OrCAD/SDT) is a complete and flexibile schematic capture package. Easy to use menu driven commands help you to create, edit, save and plot elctronic schemtics.

Developed specifically to run on IBM personal computers and compatibles, OrCAD/SDT supports most of the popular graphics boards, printers, and plotters. This elminates the need for special, proprietary hardware by enabling you to use standard output equipment.

1.1. FEATURES OF OrCAD/SDT

The OrCAD/SDT software package consists of the schematic drafting program DRAFT, a library de-compiler called DECOMP, a library compiler called COMPOSER, Net List, Design Check, and other utility programs. Outlined below is a summary of each program supplied with your OrCAD/SDT.

1.1.1. DRAFT

DRAFT is the schematic drafting program that enables you to create, edit and save schematic worksheets. The major features of DRAFT include:

* Over 3000 Unique library parts
* DeMorgan Equivalnt parts
* Placement of wires, buses, connectors, labels and junctions.
* Real-time rubberbanding of wires and buses when objects are moved
* Part rotation and mirroring
* Moving, replicating and deleting objects or blocks of objects
* Powerful step-and-repeat command
* Visible grid dots and angled bus entries
* Automatic panning of the worksheet
* Five zoom levels
* Over 100 user-assignable macros
* Unlimited levels of hierarchy
* On-Line part browsing and library directory
* String searching
* Vertical text placement
* Suspension of session for DOS command execution
* Supports "A" through "E"-size worksheets

1.1.2. Part Libraries

Included with OrCAD/SDT are extensive part libraries of the most commonly used devices in the industry. These libraries include TTL, CMOS, memory, ECL, discrete, analog, microprocessor,

and peripheral devices.

1.1.3. DECOMP

DECOMP is a library de-compiler that converts the OrCAD-supplied library object files (files with a .LIB extension) to library source files. You cn then edit the source files using a standard text editor (OrCAD's Symbol Description Language enables you to create simple or complex symbols) to make additions or modfications to the library.

Using COMPOSER, you then compile the edited custom files for loading into DRAFT.

1.1.4. COMPOSER

COMPOSER is a library compiler that converts your custom library source files into the highly compressed library object files used by DRAFT. After COMPOSER cmpiles a source fil into an object file, your custom library is ready for loading into DRAFT and the other utilities.

1.1.5. Other Utility Programs

OrCAD/SDT's flexibility continues after the schematic design process with easy-to-use utility programs, including:

TREELIST:

A program that scans a hierarchical organiztion of sheet to display the structure, sheet names, and sheet path names of the hierarchy.

ANNOTATE:

This program scans a hierarchy or flat file and automtically updates all part reference designators (U?, R? etc. explained later).

ANNOTATE also updates the pin numbers associated with the reference designators. ANNOTATE can handle very large, complex and multiple worksheets.

PRINTALL:

PRINTALL prints a schematic sheets, including a hierarchy, flat file, or annotation file, in batch mode.

PLOTALL:

PLOTALL plots a schematic or group of schematic sheets, including a hierarchy, flat file, or annotation file, in batch mode.

PARTLIST:

This utility summarizes all the parts used in a schematic or group of schematic sheets. You can also merge user-specific information with the PARTLIST report.

ERCHECK:

This is a utility that performs an electrical rules check of your schematic worksheets. The program checks for shorts, inputs with no driving source, and other common errors.

NETLIST:

This program generates a netlist if the worksheet signal and part connections in these formats: Algorex, Applicon (Leap and BRAVO !) Cadnetix, Calay, Computervision, EDIF, Future Net (Pinlist and Netlist), Multiwire, PCAD, PSpice, Racal-Redac, Sicards (old), and Telesis formats. NETLIST also generates a general wire list.

BACKANNO:

The BACKANNO utility updates part reference designtors in your design. The input to the program, list of old and new reference designtors, is used to update your schematic worksheets.

CLEANUP:

This utility checks the worksheet for wires, buses, junctions, labels, modul ports, and other objects that are placed on top of each other. CLEANUP removes duplicate or overlapping wires, buses, and junctions, and displays warning mssage advising you of other duplicate objects.

1.2. HOW TO USE THIS MANUAL

This manual provides first time and experienced users with a complete, easy-to-use reference guide and tutorial. First time users are advised to read Sections 2-6. Experienced users may need to refer to Section 4 and 6 for a review of the DRAFT command set and the utilities.

Users who wish to create their own "custom" part libraries should read Section 7. Section 8 is a step-by-step introductory tutorial recommended for all users.

Section 2:

Describes the installation and configuration of OrCAD/SDT on your personal computer.

Section 3:

Gets you started by describing how to invoke DRAFT and how to select commands from the pop-up menus. You will also learn how to save, load, print and plot fields.

Section 4:

Describes the DRAFT command set.

Section 5:

Discusses DRAFT's hierarchical features, and explines both simple and complex hierarchies.

Section 6:

Shows you how to use the utility programs included with OrCAD/SDT.

Section 7:

Introduces COMPOSER and DECOMP and outlines the procedure for creating your own libraries. This section concludes with instructions for creating your own library parts.

Section 8:

A detailed tutorial discussing the use of the DRAFT command set and uilities. The tutorial creates a worksheet, where you will learn many of the concepts used in schematic capture.

Appendix:

Lists all of the library parts included with DRAFT. Outlines the configurations of the supported graphics boards, and lists the printer, plotter, and graphics board drivers supplied.

Index:

A complete Index to the manual.

1.3. TROUBLESHOOTING

If you have questions regading how to run the program, be sure that you have properly configured your system. Check to see that you have the correct driver for the graphics board installed in your system. See Section 2 and the Appendix for configuration information and for graphics board jumpering information.

Be sure that your are using the correct printer or plotter driver. If you have a printer or plotter problem, see the PRINTALL and PLOTALL utility program information in Section 6.

If you have a computer or mouse problem, consult your user's manual or contact your dealer.

1.3.1. If You Have Questions

If you need help after you hve checked the items discussed above, we may be able to assist you. Before you contact OrCAD for technical support, be prepared to give the support engineer the following infrormation:

1. The name of the Licensed User of your OrCAD/SDT.
2. Registration number.
3. Program Version Number
4. Type of personal computer.
5. Size of RAM in personal computer.
6. Types of peripheral boards installed.
7. Printer or plotter name and model number.
8. Configuration of your OrCAD/SDT System (obtained from the main menu when you invoke "DRAFT/C").

The OrCAD Telephone number is: 1.503.640.5007.Ask for Technical Support.

N O T E

The person who fills out and sends in the registration card is the only one entitled to customer support. If you purchased OrCAD/SDT and sent in the warranty registration card with 30 days, you will receive one year of free customer support.

At OrCAD System Corporation, we are committed to giving our customers the finest products and support. Before you free support period expires, be sure you apply for the OrCAD System Corporation Continued Support Plan. Contact OrCAD System Corporation for information.

.pn13

2. SETTING UP OrCAD/SDT

Now that you have been introduced to OrCAD/SDT's DRAFT and its utility programs, you will learn how to install and cofigure these programs on your system.

In this section, you'll learn the procedure for masking backup copies of your OrCAD/SDT program diskettes, and you'll install OrCAD/SDT on your dual-floppy or hard disk system. You will configure the software by loading graphics, printer, and plotter drivers for your hadware, and the symbol libraries for your designs. The section concludes by showing you sample configuration examples.

2.1. WHAT YOU NEED TO BEGIN

Let's review the items that you'll need to install OrCAD/SDT. These items are supplied with your OrCAD/SDT software package:

1. Software packet containing these three OrCAD disks:

   * MASTER SOFTWARE DISK
   * UTILITY SOFTWARE DISK
   * LIBRARY DISK
   * DRIVER DISK/DISPLAY
   * DRIVER DISK/PRINTER PLOTTER

2. OrCAD Schematic Design Tools documentaion.

You need to supply:

1. An IBM PC/XT/AT or compatible personal computer with two double-sided, double density 360 K floppy disk drivers or a hard disk system.

2. DOS Version 2.0 or greater.

3. A minimum of 256 K bytes of PC memory.

4. Five formatted, double-sided disks for making backup copies of the OrCAD supplied progrms.

5. Four extra formatted, double-sided disks, if you are installing OrCAD/SDT on dual-floppy system.

2.2. PREPARING FOR INSTALLATION

Before you install OrCAD/SDT on your system, you should be familiar with the DOS commands CHDIR (CD), COPY, DIR, DISKCOPY, FORMAT and MKDIR (MD). If your system has a hard disk, you should lso familiarize yourself with tree-structured directories. For a description of these commands and features, see your DOS user's manul.

2.2.1. Making Bckup Copies

To protect your software investment, we recommend that you make a backup copy of your program diskettes. You may make backups as detailed in the License Agreement. Opening the sealed software package and using the program constitutes acceptance of the Agreement.

If your original OrCAD/SDT program diskettes become damaged, OrCAD will supply replacements on a one-for-one basis. Send each original, damaged diskette and a check for S25 to OrCAd for a replacement.

To make backup copies of your OrCAD/SDT program diskettes, use the DOS COPY or DISKCOPY commands. See your DOS user's manual for instructions.

2.3. INSTALLING OrCAD

OrCAD/SDT may be installed on a dual-floppy or hard disk system. If you are installing OrCAD/SDT on a system having two floppy drivers, see Section 2.3.1. If you are installing OrCAD/SDT on a hard disk system, see Section 2.3.2.

After you have installed OrCAD/SDT, see Section 2.4. for information on configuring your system.

2.3.1. Installing OrCAD/SDT on a Dual Floppy System

If your system has two floppy disk drives, you need four formatted, double-sided disks to make working copies of OrCAD/SDT. These working copies are needed to load the drivers, libraries, and worksheets filenames. Since the files do not all fit on a single disk, separating them according to the procedure suggested below keeps disk swapping to a minimum.

Label four formatted floppy disks as follows:

* DRAFT
* DRIVER/LIBRARY
* SHEET
* LIBRARY UTILITY

Copy the executable and overlay files from your OrCAD supplied MASTER SOFTWARE DISK and UTILITY SOFTWARE DISK (except for COMPOSER, DECOMP, TREELIST, and CLEANUP) to your working disk DRAFT. Executable files have the .EXE extension after the filename and overlay files use .OVL.

Copy the TREELIST and CLEANUP executable files (.EXE files), the DRAFT overlay file (.OVL file), and the drivers and libraries to your working disk DRIVER/LIBRARY.

Copy COMPOSER, DECOMP, and the OrCAD-supplied to your working disk LIBRARY UTILITY.

You will use the working disk SHEET for your worksheet files.

Figures 2-1, 2-2 and 2-3 show the program files and their

working disk locations. Follow the 10-step procedure below for installing OrCAD/SDT on dual-floppy system.

Files Required	Where Found
COMMAND.COM	your DOS Disk
DRAFT.EXE	MASTER SOFTWARE DISK
DRAFT.OVL	MASTER SOFTWARE DISK
ERCHECK.EXE	UTILITY SOFTWARE DISK
ANNOTATE.EXE	UTILITY SOFTWARE DISK
NETLIST.EXE	UTILITY SOFTWARE DISK
BACKANNO.EXE	UTILITY SOFTWARE DISK
PARTLIST.EXE	UTILITY SOFTWARE DISK
PRINTALL.EXE	UTILITY SOFTWARE DISK
PLOTALL.EXE	UTILITY SOFTWARE DISK

Figure 2-1. Required DRAFT

Files Required	Where Found
DRAFT.OVL	MASTER SOFTWARE DISK
TREELIST.EXE	UTILITY SOFTWARE DISK
CLEANUP.EXE	UTILITY SOFTWARE DISK
PRINTER.DRV	DRIVER DISK/PRINT.-PLOT.
Required Printer Driver	DRIVER DISK/PRINT.-PLOT.
Required Plotter Driver	DRIVER DISK/PRINT.-PLOT.
Required Graphics Card Driver	DRIVER DISK/DISPLAY
Library Files (.LIB extension)	LIBRARY DISK

Figure 2-2. Required DRIVER/LIBRARY Disk Files

Files Required	Where Found
COMPOSER.EXE	MASTER SOFTWARE DISK
DECOMP.EXE	MASTER SOFTWARE DISK
Library Files (.LIB extension)	LIBRARY DISK

Figure 2-3. Required LIBRRY UTILITY Disk Files

STEP 1:

Insert your DOS into drive A, a formatted disk labeled DRAFT into drive B, and at the DOS prompt, type:

```
COPY A:COMMAND.COM B:<ENTER>
```

COMMAND.COM is copied from drive A to the DRAFT disk in drive B.
Remove the DOS disk from drive A.

STEP 2:

To copy the executable and overlay files to the DRAFT disk in drive B, insert the OrCAD-supplied MASTER SOFTWARE DISK into drive A, and at the DOS prompt type:

COPY A:DRAFT.* B:<ENTER>

Remove the MASTER SOFTWARE DISK from drive A.

STEP 3:

Insert the OrCAD-supplied UTILITY SOFTWARE DISK into drive A, copy the following utility programs to the DRAFT disk in drive B. At the DOS prompt type:

COPY A:ERCHECK.EXE B:<ENTER>

Then type:

COPY A:ANNOTATE.EXE B:<ENTER>

Then type:

COPY A:NETLIST.EXE B:<ENTER>

Then type:

COPY A:BACKANNO.EXE B:<ENTER>

Then type:

COPY A:PARTLIST.EXE B:<ENTER>

Then type:

COPY A:PRINTALL.EXE B:<ENTER>

Then type:

COPY A:PLOTALL.EXE B:<ENTER>

Copies of the files shown in Figure 2-1 should now be on the DRAFT disk in drive B.
Remove the DRAFT disk in drive B.

STEP 4:

Insert a blank, formatted disk labeled DRIVER/LIBRARY into drive B. Copy the remaining utilities from the OrCAD-supplied UTILITY SOFTWARE DISK to the DRIVER/LIBRARY disk in drive B. At the DOS prompt, type:

COPY A:TREELIST.EXE B:<ENTER>.

Then type:

COPY A:CLEANUP.EXE B:<ENTER>

Remove the UTILITY SOFTWARE DISK from drive A.

STEP 5:

Insert the OrCAD-supplied MASTER SOFTWARE DISK into drive A. Copy the DRAFT overlay file to the DRIVER/LIBRARY disk in

drive B. Type:

COPY A:DRAFT.OVL B:<ENTER>

Remove the MASTER SOFTWARE DISK from drive A.

STEP 6:

Turn to Appendix B and locate the file names for the graphics, printer, and plotter drives used in your computer.

Insert the OrCAD-supplied DRIVER DISK/DISPLAY into drive A.

Copy the graphics driver file from the DRIVER DISK/DISPLAY to the DRIVE/LIBRARY disk in drive B. Type:

COPY A:Filename.DRV B:<ENTER>

For example, if your computer has an IBM EGA graphics card installed, and you want the driver with 640 x 350 16-color resolution, type:

COPY A:EGA16E.DRV B:<ENTER>

The graphics driver is copied to the DRIVER/LIBRARY disk in drive B. Remove the OrCAD-supplied DRIVER DISK/DISPLAY from drive A.

STEP 7:

Insert OrCAD-supplied DRIVER DISK/PRINTER-PLOTTER into drive A. Copy the printer driver (EPSON.DRV, TOSHIBA.DRV etc.) and plotter driver (HI.DRV, HP.DRV etc.) that corresponds to he printer or plotter that you have, to the DRIVER/LIBRARY disk in drive B. For example, if you have an EPSON FX100 printer, type:

COPY A:EPSON.DRV B:<ENTER>

Refer to the Appendix for the names of the printer and plotter drivers.

STEP 8:

Copy the printer driver file PRINTER.DRV from the DRIVER DISK/PRINTER-PLOTTER from drive A.

STEP 9:

Insert OrCAD-supplied LIBRARY DISK into drive A. Copy the OrCAD-supplied libraries (files with .LIB extension) from the LIBRARY DISK in drive A to the DRIVER/LIBRARY disk in drive B.Type:

COPY A:*.LIB B:<ENTER>

N O T E

You need not copy all the libraries to your DRIVER/LIBRARY disk. The library files and utility programs you need will depend on the components you spcify in your design.

Remove the LIBRARY DISK from drive A and the DRIVER/LIBRARY

disk from drive B. Your working DRIVER/LIBRARY disk is ready for use.

STEP 1:

Place the formatted disk labeled SHEET aside. You will use this disk later for storing your worksheet files.

STEP 11:

Insert the formatted LIBRARY UTILITY disk into drive B. Copy the files shown in Figure 2-3 to the LIBRARY UTILITY disk in drive B. This completes the preparation of your working disks.

Place the DRAFT working disk in drive A, and the DRIVER/LIBRARY working disk in drive B. Go to Section 2.4 and follow the procedures for configuring your OrCAD/SDT software.

2.3.2. Installing OrCAD/SDT on a Hard Disk System

The most efficient way to organize your hard disk when using OrCAD/SDT is to create a series of subdirectories. This enables you to easily organize your drivers, libraries, and worksheets.

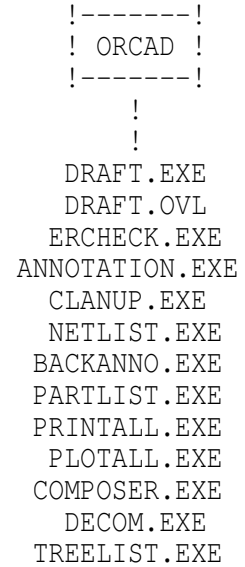
Create new directory on your hard disk called ORCAD. Within the OrCAD directory, create three more subdirectories: DRIVER, LIBRARY and SHEET.

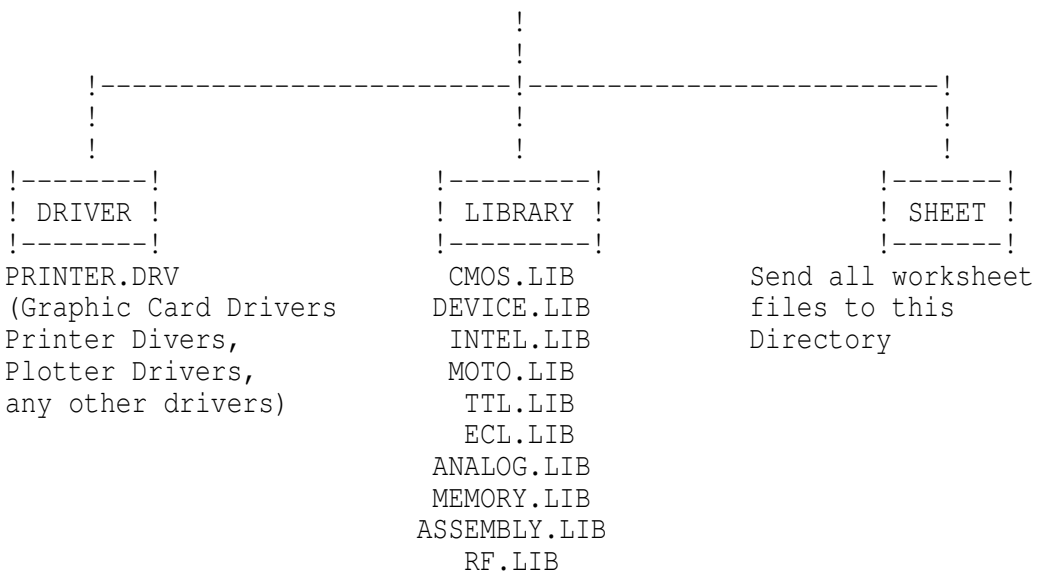
Copy the executable and overlay files from the OrCAD-supplied MASTER SOFTWARE DISK and UTILITY SOFTWARE DISK to the ORCAD directory. Next, copy files from the OrCAD-supplied DRIVER DISK/DISPLAY and the DRIVER DISK/PRINTER-PLOTTER to the DRIVER subdirectory, and the files from the OrCAD-supplied LIBRARY DISK with .LIB filename extension to the LIBRARY subdirectory.

Finally, the SHEET subdirectory is where you will put your schematic worksheet files. For now, the SHEET subdirectory remains empty.

Figure 2-4 shows the configuration tree we recommend for organizing your hard disk files. Follow STEPS 1-7 below to install OrCAD/SDT on your hard disk system.

.PA





(other user-created source and object libraries)

Figure 2-4. Recommended Hard Disk Installation

STEP 1:

Be sure that you are in the root directory. At the DOS C> prompt, create a new directory called ORCAD using the DOS command. Type:

MD ORCAD <ENTER>

Enter ORCAD, the directory you just created, by typing:

CD ORCAD <ENTER>

The DOS command CD enables you to change the current DOS directory to a new one you specify.

STEP 2:

Make three more subdirectories within ORCAD. Type:

MD LIBRARY <ENTER>

Then type:

MD DRIVER <ENTER>

Then type:

MD SHEET <ENTER>

STEP 3:

Insert the OrCAD-supplied MASTER SOFTWARE DISK into drive A. Copy the files from the MASTER SOFTWARE DISK to the OrCAD directory. Type:

COPY A:*.* C:<ENTER>

When the files are copied, remove the MASTER SOFTWARE DISK from drive A.

STEP 4:

Insert OrCAD-supplied UTILITY SOFTWARE DISK into drive A. Copy the remaining utilities to the ORCAD directory. Type:

COPY A:*.* C:<ENTER>

The files shown in Figure 2-4 should now be in the ORCAD directory. Remove the UTILITY SOFTWAE DISK from drive A.

STEP 5:

Insert the OrCAD-supplied LIBRARY DISK into drive A. Copy the library files from the LIBRARY DISK to the LIBRARY subdirectory. Type:

COPY A:*.LIB LIBRARY*.LIB <ENTER>

The Library object files (files with a .LIB extension) should now be in the LIBRARY directory. Remove the LIBRARY DISK from drive A.

STEP 6:

Insert the OrCAD-supplied DRIVER DISK/DISPLAY into drive A. Copy the files from the DRIVER DISK/DISPLAY to the DRIVER subdirectory. Type:

COPY A:*.* DRIVER*.*<ENTER>

The graphics drivers are now in the DRIVER subdirectory. Remove the DRIVER DISK/DISPLAY from drive A.

STEP 7:

Insert the OrCAD-supplied DRIVER DISK/PRINTER-PLOTTER into drive A. Copy the files from the DRIVER DISK/PRINTER-PLOTTER to the DRIVER subdirectory. Type:

COPY A:*.* DRIVER*.* <ENTER>

The printer and plotter drives are now in the DRIVER subdirectory. Remove the DRIVER DISK/PRINTER-PLOTTER from drive A. This completes the OrCAD/SDT installation on a hard disk system.

If you plan to use a mouse with your OrCAD/SDT software, see Section 2.3.3. Go to Section 2.4. and follow the procedures for configuring your OrCAD/SDT software.

2.2.3. Installing a Mouse

OrCAD/SDT supports the Microsoft and parallel mouse, the Mouse Systems serial mouse, and many compatibles. Be sure that the correct mouse driver, supplied by the mouse manufacturer, is installed and configured to he proper serial channel.

OrCAD/SDT uses the MOUSE.COM driver for the Microsoft Mouse and the MSMOUSE.COM driver for the Mouse System Mouse. Both of

these driver names use the /1 or /2 option switches for configuring them for use on the proper (COM1 or COM2) serial channel.

OrCAD/SDT has no spcial mouse requirements. If you have mouse problems, see your mouse user's guide for information.

2.4. CONFIGURING OrCAD/SDT

After you have installed the OrCAD/SDT files on your system, you must configure the software to access the correct printer, plotter and graphics drivers, and libraries.

.PN22

The configuration information that you supply will vary depending on whether you are using floppy or hard disk system. Example are given below for each configuration case.

The procedures outlined in this section show you how to configure these items:

* Driver Prefix
* Display Driver
* Printer Driver
* Plotter Driver
* Library Prefix
* Library Files
* Worksheet Prefix
* Macro File
* Initial Macro
* Macro Buffer Size
* Hierarchy Buffer Size
* Color Table/Pen Plotter Table

2.4.1. Invoking the OrCAD/SDT Configuration File

To configure OrCAD/SDT, you must be in the ORCAD directory if you are using a hard disk system. The ORCAD directory must contain the files DRAFT.EXE, DRAFT.OVL and the utility programs (see Figure 2-4). If yours is a dual floppy system, place the DRAFT working disk in drive A and the DRIVER/LIBRARY disk in drive B.

At DOS prompt, type:

DRAFT /C <ENTER>

/C is a switch that causes the configuration menu to be displayed on the screen. Use the /C switch when you want to change the OrCAD/SDT's configuration.

When invoked, the "Configuration of OrCAD/SDT" menu appears on the screen as shown in figure 2-5. Perform each of the steps described in sections 2.4.2-2.4.16 below.

```
!-------------------------------------------------!
!    :::  CONFIGURATION OF OrCAD/SDT  :::         !
!                                                 !
!    DP - Driver Prefix                           !
!    DD - Display Driver                          !
!    PD - Printer Driver                          !
!    PL - Plotter Driver                          !
!    LP - Library Prefix                          !
```

```
!    LF - Library Files                          !
!    WP - Worksheet Prefix                       !
!    MF - Macro File                             !
!    IM - Initial Macro                          !
!    MB - Macro Buffer Size            16384     !
!    HB - Hierarchy Buffer Size        1024      !
!    CT - Color Table/Plotter Pen Table          !
!    U  - Update Configuration Information       !
!    Q  - Quit, Abandon to DOS                   !
!    R  - Run Program                            !
!    Command ?                                   !
!------------------------------------------------!
```

Figure 2-5. The Configuration Menu

2.4.2. Driver Prefix

Driver prefix sets the subdirectory path or disk drive enabling DRAFT to load the printer, plotter, and graphics drivers.

To select driver prefix, type DP. At the ">>>>>>" prompt, type the subdirectory path or disk drive letter where the driver files are located, then press <ENTER>.

EXAMPLES:

DP - Driver Prefix
>>>>>>B: (Looks for the drivers in Drive B. Use this prefix if you have a dual-floppy sistem)

DP - Driver Prefix
>>>>>>DRIVER\ (Looks for the drivers in the DRIVER subdirectory on hard disk systems. Be sure that the back slash "\" is present)

2.4.3. Display Driver

Display driver configures OrCAD/SDT for your sistem's graphics driver.

To configure OrCAD/SDT for your graphics driver, type DD. A table shows a list of the supported graphics drivers. Type the letter that corresponds to your graphics driver. If yours is not listed, type S and enter your driver name. Type Q to quit and return to the configuration menu.

2.4.4. Printer Driver

Printer driver configures OrCAD/SDT with the printer driver your printer uses.

To configure OrCAD/SDT for your printer driver, type PD, then select one of the printer drivers from the table by typing the corresponding letter.

Type X if none of the drivers are to be used, then type Q to return to the main configuration menu. Type an S if you are using a special driver not listed. Enter the driver name at the '>>>>>>" prompt.

2.4.5. Plotter Driver

Plotter driver configures OrCAD/SDT with the plotter driver

used with your plotter. You may also specify serial channel number, baud rate, parity, and word length.

To configure OrCAD/SDT with your plotter driver and the correct serial channel information, type PL. A menu displays a list of supported plotter drivers. Type the letter that corresponds to the appropriate driver.

When you have selected a plotter driver, type a colon (:) to configure DRAFT for serial channel 1 or 2. Next, you may select the baud rate, parity, and word length your plotter needs for data transfer.

NOTE
Be sure that the serial channel, baud rate, parity and word length match the characteristics of your plotter. See your plotter user's manual if you need more information.

Type X if no drivers are used. Type an S if you are using a special driver not listed. Enter the driver name at the '>>>>>>' prompt. To quit and return to the configuration menu, type Q.

For more plotter information, see the PLOTALL utility in Section 6.

2.4.6. Library Prefix

Library prefix sets the subdirectory path or disk drive that enables DRAFT to load the part libraries.

To select the library prefix, type LP. At the '>>>>>>' prompt, type the subdirectory path or disk drive letter where the library files are located. Press <ENTER>.

EXAMPLES:

LP - Library Prefix
>>>>>>B: (Looks for the libraries in drive B. Use this prefix if you have a dual floppy system)

LP - Library Prefix
>>>>>>LIBRARY\ (Looks for the libraries in the LIBRARY subdirectory on hard disk systems)

2.4.7. Library Files

To configure DRAFT with the part libraries, type LF. The screen displays a list of additional selections.

To add a library, select A. When "Enter New Name>>>>>>" appears, type the name of your part library selection and press <ENTER>. If you enter more than one library, "Add File Name Before? [A...B] appears. Type the letter to enter the library name in the order that you have selected.

EXAMPLE:

```
::: Library File :::
       DEVICE.LIB
       TTL.LIB
       MEMORY.LIB
```

The three part libraries listed above are loaded by DRAFT

and many of the utility programs.

To remove a library, type R. When "Remove Which File Name?" appears, enter the letter that represents the library you want removed.

Type Q to quit and return to the main configuration menu.

2.4.8. Worksheet Prefix

Worksheet prefix sets the subdirectory path or disk drive specifier that enables DRAFT to load the worksheet files.

To select worksheet prefix, type WP. When ">>>>>>" appears, type the subdirectory path or disk drive specifier where you want worksheet files stored, then, press <ENTER>.

.pa

EXAMPLES:

WP - Worksheet Prefix >>>>>>B:	(On dual-floppy systems, select drive B)
WP - Worksheet Prefix >>>>>SHEET	(On hard disk systems, sends worksheet files to the SHEET subdirectory)

2.4.9. Macro File

Macro file enables you to automatically load a macro file when DRAFT is invoked. To load a macro file, type MF. At the '>>>>>>' prompt, type the path and name of the desired macro file.

EXAMPLES:

>>>>>>A:MACRO.MAC	(On dual-floppy system, looks for the MACRO.MAC file in drive A)
>>>>>>MACRO.MAC	(On a hard disk system, enter MACRO.MAC)

2.4.10. Initial Macro

Initial macro enables you to execute a specific macro automatically when DRAFT is invoked. To configure an initial macro, type IM. At the ">>>>>>" prompt, enter the ASCII character name of the macro you want to run, then press <ENTER>

EXAMPLES:

If you want to assign the F1 key to execute a macro, press the 'F' key then the '1' key, followed by <ENTER>.

If you want to assign Ctrl-A to execute a macro, type ^ (shift 6), then A. Then press <ENTER>.

NOTE

To execute an initial macro, the <ENTER> key must be the first key you press when creating the macro. This allows the program to move past the "Load File" prompt and the

copyright screen when DRAFT is invoked. The macro must also be saved and loaded as a macro file (see Section 2.4.9).

For exemple:

To create an initial macro (using the <F6> key) that turns on grid dots, the macro format is as follows (See section 4.11.12 for a discussion on macro formats):

{F6} = {ENTER}sgvy{}

2.4.11. Macro Buffer Size

This selection enables you to modify the amount of memory allocated to the macro buffer, the storage location for the macros. The minimum memory size for the macro buffer is 16384 bytes; the maximum is 65535.

Type MB to change the macro buffer size. At the "Macro Buffer Size?" prompt, enter the desired buffer size. The number must be in the range 16948 - 65535.

2.4.12. Hierarchy Buffer Size

Hierarchy buffer size enables you to modify the amount of memory allocated to the hierarchy buffer, used to store all hierarchical sheet and path names. The minimum memory size for this buffer is 1024 bytes, which enables you to create a hierarchical depth of about 75-100 worksheets (depending on sheet and pathname character lengths). You can increase the size of the hierarchy buffer to 65535 bytes, large enough for a hierarchical depth of up to 5000 worksheets. Type HB to change hierarchy buffer size. At the "Hierarchy Buffer Size?" prompt, type the desired buffer size. The number must be in the range 1024-65535.

2.4.13. Color Table/plotter Pen Table

Color table/plotter pen table enables you to modify the screen display and plotter colors for library parts, pin numbers and names, wires, buses, junctions, connectors, and other objects in the worksheet.

Type CT to display the color table/plotter pen table.

To modify the color table parameters for screen display, type a letter that corresponds to the object that you want to change. From the color table right-hand column, enter the letter for the color desired at the Enter Letter for the new color>" prompt.

To select the appropriate plotter pen color, type P, followed by the alpha character (A-O) at the "Enter Pen Number" prompt. If you enter 99, the plotter will not plot the object. If you enter 0 (zero), the plotter pauses, enabling you to change pens.

Typing V (velocity select) followed by the alpha character A through O, enables you to change the velocity of the plotter pen. See your plotter user's manual to correctly set this commpand, or leave it in the DEFAULT configuration.

Type Q to quit and return to the main configuration menu.

See "Four-Color Mode Configuration" in the Appendix for selecting colors on graphic boards in the four-color mode.

EXAMPLES:

To suppress the worksheet border and title block when plotting, type PM, followed by 99 <ENTER>.

To change the velocity of the pen used for drawing buses, type VE, followed by the value specified in your plotter user's manual.

2.4.14. Update Configuration Information

Update configuration information, as the title implies, enables you to write the latest configuration information (the procedure we have discussed in this section) to DRAFT. After you update the configuration information, it is automatically loaded when you invoke DRAFT or the utilities.

To update the configuration information, tupe U. Once the DRAFT configuration is updated, the main configuration menu remains on the screen.

CAUTION ! If you quit the program before updating the configuration information, all information is lost.

2.4.15. Returning to DOS with Quit

Quit enables you to return to DOS without updating the configuration information.

Type Q to quit the program and abandon to DOS. If ant changes were made to the configuration menu, the program first asks if you want to update the configuration information before returning to DOS.

2.4.16. Run Program

Run program enables you to invoke DRAFT with the latest configured information that appears on the configuration menu.

Type R to invoke DRAFT or a utility program.

2.4.17. Dual-Floppy Configuration Example

Figure 2-6 below, shows OrCAD/SDT configured for a dual-floppy system.

```
!-----------------------------------------------!
!    :::  CONFIGURATION OF OeCAD/SDT :::        !
!    DP - Driver Prefix                         !
!              B:                               !
!    DD - Display Driver      EGR16E.DRV        !
!    PD - Printer Driver      EPSON.DRV         !
!    PL - Plotter Driver      HI.DRV            !
!    LP - Library Prefix                        !
!              B:                               !
!    LF - Library Files                         !
!              CMOS.LIB       TTL.LIB    RF.LIB !
!              DEVICE.LIB     ECL.LIB           !
!              INTEL.LIB      MEMORY.LIB        !
!              MOTO.LIB       ASSEMBLY.LIB      !
!    WP - Worksheet Prefix                      !
!              B:                               !
!    MF - Macro File                            !
!              A:MACRO.MAC                      !
```

```
!    IM - Initial Macro                          !
!    MB - Macro Buffer Size        16384         !
!    HB - Hierarchy Buffer Size    1024          !
!    CT - Color Table/Plotter Pen Table          !
!    U  - Update Configuration Information       !
!    Q  - Quit, Abandon to DOS                   !
!    R  - Run Program                            !
!    Command ?                                   !
!------------------------------------------------!
```

Figure 2-6. Sample Dual Floppy Configuration

The driver and library prefixes in this example are configured for drive B, since the DRIVER/LIBRARY disk is placed in drive B before invoking DRAFT. The worksheet prefix is also configured for drive B.

After invoking DRAFT, you must remove the DRIVER/LIBRARY disk from drive B at the "Load File?" prompt. Then, insert the SHEET disk in drive B, enabling DRAFT to load your worksheet files.

NOTE

You must configure the TREELIST and CLEANUP utility programs located on the DRIVER/LIBRARY disk when you invoke them for the first time.

If you want to run the DRIVER/LIBRARY disk in drive A to use TREELIST or CLEANUP, the drive specifiers for the driver and library files must be for drive A. Likewise, if you want to run the DRIVER/LIBRARY disk in drive B, the drive specifiers for the driver and library files must be for drive B.

.pa

2.4.18 Sample Hard Disk Configuration

Figure 2-7 shows OrCAD/SDT configured for a hard disk system.

```
!------------------------------------------------!
!    :::  CONFIGURATION OF OeCAD/SDT :::         !
!    DP - Driver Prefix                          !
!              DRIVER\                           !
!    DD - Display Driver      EGR16E.DRV         !
!    PD - Printer Driver      EPSON.DRV          !
!    PL - Plotter Driver      HI.DRV             !
!    LP - Library Prefix                         !
!              LIBRARY\                          !
!    LF - Library Files                          !
!              CMOS.LIB       TTL.LIB    RF.LIB !
!              DEVICE.LIB     ECL.LIB            !
!              INTEL.LIB      MEMORY.LIB         !
!              MOTO.LIB       ASSEMBLY.LIB       !
!    WP - Worksheet Prefix                       !
!              SHEET\                            !
!    MF - Macro File                             !
!              MACRO.MAC                         !
!    IM - Initial Macro                          !
!    MB - Macro Buffer Size        16384         !
!    HB - Hierarchy Buffer Size    1024          !
!    CT - Color Table/Plotter Pen Table          !
```

```
!    U  - Update Configuration Information      !
!    Q  - Quit, Abandon to DOS                  !
!    R  - Run Program                           !
!    Command ?                                  !
!-----------------------------------------------!
```

Figure 2-7. Sample Hard Disk Configuration

The driver, library, and worksheet prefixes have been configured as DRIVER\, LIBRARY\ AND SHEET\ respectively. This corresponds to the like-named subdirectories you created within the ORCAD directory.

2.5. IF YOU HAVE CONFIGURATION PROBLEMS

The most common configuration problem that you will likely encounter is: MSDOS Error # 2 File Not Found. If you see this error message, check for these conditions:

1. If you are using a hard disk, be sure that there is a back slash (\) after DRIVER\, LIBRARY\, and SHEET\ in their respective prefixes.

2. If yours is a dual-floppy system, be sure that you have specified b: for the driver, library, and worksheet drive prefixes.

Another common error is: MDOS Error # 3 Path Not Found. If you see this error message, be sure that there are not blank spaces preceding or following the name of a text entry.

For example, there should be no space preceding or following DRIVER\, TTL.LIB, or any other text entry, Re-enter any questionable entries.

2.6. SEND IN YOUR REGISTRATION CARD

Remember to send in your product registration card, located in the sealed packed that contained your OrCAD/SDT program diskettes. Only by sending in your registration card do you become eligible for product updates.

}PDRAFT}.pn32

3. GETTING STARTED

Section 1 introduced OrCAD/SDT, presenting its major features.

Section 2 showed you how to configure OrCAD/STD to run your personal computer.

In this section you will:

* Learn to invoke DRAFT and execute keyboard and mouse commands.

* Receive an overview of the function that make OrCAD/STD easy to use: loading, saving and printing worksheet files.

* Be introduced to the three types of file structures used for creating your worksheets: flat file, hierarchy, and one-sheet file structures.

For detailed command description, see Section 4. If you are a first time user, we recommend that you review the tutorial in Section 8, after you read this section.

3.1. INVOKING DRAFT

When you are configured DRAFT for the libraries, graphics board, printer, and plotter drivers, the program is ready to run. To invoke DRAFT, be sure that you are in the DRAFT subdirectory if you are using a hard disk system. If you are using a dual floppy disk system, place the DRAFT disk that you created in Section 2 into drive A, and the DRIVER/LIBRARY disk into drive B.

Type the following at the DOS promt:

DRAFT <ENTER>

After DRAFT loads the drivers and libraries, a screen appears displaying the OrCAD logo. At the top of the screen, "Type any key to continue" appears. Type any key to display the Copyright Notice screen.

In a few seconds, "Type any key to continue" is repetead at the top of the screen. Typing any key displays: "Load File?".

You now enter a filename, or begin a new worksheet as follows:

To begin a new worksheet, press <ENTER> when the "LOAD FILE?" promt is displayed.

To load a previously created worksheet, enter the filename and press <ENTER>.

.pa

3.2. COMMAND AND MENU ORGANIZATION

DRAFT is an interactive, schematic capture program that uses pop-up command menus and prompts. The commands are further categorized into main command and subcommand menus. You can display both commmand menus on the screen by pressing <ENTER> ar by using a mouse. Figure 3-1 ilustrates the main command menu as it appears on the screen.


```
-----------------------------------------------------------
| Again         |
| Block         |
| Conditions    |
| Delete        |
| Edit          |
| Find          |
| Get           |
| Hardcopy      |
| Jump          |
| Library       |
| Macro         |
| Place         |
| Repeat        |
| Quit          |
```

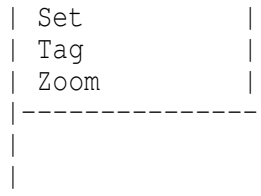


Figure 3-1 The Main Command Menu

If you are at the main command menu (you have not yet executed a command) and you press <ENTER>, the main command menu is displayed on the screen. If you are at a subcommand menu (you have executed a command) and you press <ENTER>, the subcommand menu is displayed on the screen. A subcommand menu has its main command name displayed on the prompt line (see Figure 3-2).

3.2.1. Invoking Commands

You may execute commands in two ways.

1. Press the first letter of the command name. It is not necessary for the command to be displayed on the screen.

2. Select the command or subcommand from the menu by moving the highlighted bar over it and presing <ENTER>.

A subcommand menu is shown in Figure 3-2. Notice the word Place above the menu. This tells you that the Place command has been invoked, and you may select any of these subcommands: Wire, Bus, Junction, Entry, Label, Module Port, Power, Sheet, or Dashed line.

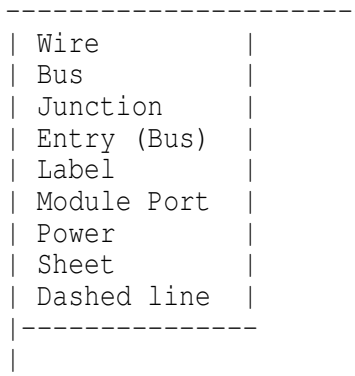
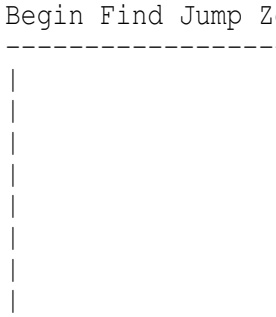


Figure 3-2 The Place subcommand Menu

Many of DRAFT's main commands have more than one level of subcommand. If you select the Wire subcommand (press W) from the Wire subcommand menu for example, Figure 3-3 shows the subcommands that are displayed.



|

Figure 3-3 The Wire Subcommands

When execute one of these commands (shown on the prompt line) by pressing the first letter of the command name, or by pressing <ENTER> to pull up the menu. The prompt line displays subcommands or user messages. You may execute any of the PLACE Wire subcommands from the prompt line.

To return the command or to abandon a command, press <ESCAPE>.

3.3. CREATING A NEW WORKSHEET

Within DRAFT, you may clear or create a new worksheet by selecting OUIT Initialize.

Once selected (if there are objects placed in the worksheet), DRAFT returns "Are you sure?" on the subcommand menu. This reminds you not in accidentally clear the worksheet. Select [No] to abandon QUIT Initialize and return to the main command level. Select [Yes] to clear the worksheet.

The new schematic worksheet created defaults to an "A" size worksheet. You may change the worksheet size by selecting "SET Worksheet Size".

3.4. LOADING WORKSHEET FILES

There are several ways to load a previously saved worksheet into DRAFT. You may load a worksheet from DOS, or after you have invoked DRAFT. You may also load worksheet from within the program editing sesion. The methods for loading a worksheet file are outlined below.

3.4.1. From DOS

From DOS you may load a worksheet when invoking DRAFT by typing:

DRAFT filename <ENTER>

Where filename is the filename of a worksheet to be loaded.

After DRAFT loads drivers and libraries, a screen appears displaying the OrCAD logo. At the top of the screen, the message "Type any key to continue" appears. Press any key to display the Copyright Notice screen.

In a few seconds, "type any key to continue" repeats at the top of the screen. When you type any key, the filename you specified on the DOS command line is loaded.

3.4.2. After invoking DRAFT

You may load a worksheet after invoking DRAFT by typing:

DRAFT <ENTER>

After DRAFT loads drivers and libraries, a screen appears displaying the OrCAD logo. At the top of the screen, the message

"Type any key to continue" appears. Press any key to display the copyright notice screen.

In a few seconds, "type any key to continue" is repeated at the top of the screen. When you press any key, the message "Load File?" is displayed. You may now enter a filename, or begin a new worksheet as follows:

* To load a previously created worksheet, enter the filename and press <ENTER>.

* To begin a new worksheet, press <ENTER> when the "Load File?" prompt is displayed.

3.4.3. From Within DRAFT

To load a worksheet from within DRAFT, select QUIT Initialize. If these are the objects placed in the worksheet, DRAFT returns "Are you sure?" on the subcommand menu. This reminds you not to accidentally clear the worksheet. Select [No] to abandon Quit Initialize and return to the main command level. Select [Yes] to clear the worksheet and load another worksheet file.

With a clear worksheet, DRAFT returns "Load File?" on the prompt line. Type in the path and the filename that you wish to load and press <ENTER>. If the filename exists, the worksheet is loaded and displayeded on the screen. If the filename does not exist, <<<new worksheet>>> appears on the prompt line.
.pa
3.5. SAVING WORKSHEETS TO A FILE

To QUIT Write command enables you to save the current worksheet to any file you specify. When you invoke QUIT Write, DRAFT returns "Write to File?" on the prompt line.

Type the desired path and filename, then press <ENTER>. The worksheet is saved to the file specified and DRAFT returns to the Quit subcommand menu.

3.6. UPDATING FILES

The QUIT Update command updates current version of the worksheet. If the current worksheet had been previously loaded from a file, that file is updated.

If the current worksheet is unnamed, DRAFT responds with "Write to File?" on the prompt line. Type is a filename and press <ENTER>.

3.7. EXITING DRAFT

To exit DRAFT and return to DOS without saving changes, select the "QUIT Abandon" command. If objects have been placed in the worksheet since the last update. DRAFT returns "Are you sure?" on the subcommand menu. Select [No] to abandon the subcommand. Select [Yes] to quit DRAFT without saving changes and return to the operating system.

3.8. PRINTING A FILE

To print a worksheet, use HARDCOPY. If you prefer to print from DOS use the PRINTALL utility.

To print, select HARDCOPY. be sure that the "HARDCOPY Destination" you select is "LPT", then choose the appropriate paper widh using the HARDCOPY Widh of Paper command. Next, select Scale or Compressed mode with the HARDCOPY print Mode command (the default is compressed print). To print, select the HARDCOPY Make Hardcopy command.

Refer to the HARDCOPY command outlined in Section 4 or the PRINTALL utility for addintional information.

3.9. PLOTTING A FILE

The PLOTALL utility plots yours files. See Section 6 for information on the PLOTALL utility.
.pa
3.10. USING A MOUSE

When you use a mouse, commands and subcommands are entered with the left and right mouse buttons. The left is the <ENTER> button and the right is unused and may be defined as a command macro key.

From the main command level, execute a command with the mouse by pressing <ENTER> (left button). This displays the menu on the screen. Move the highlighted bar to the desired command and press <ENTER> again to execute the command. Repeat this procedure to execute commands.

The DRAFT Again command, was designed for use with the mouse. It enables you to repeat the previous main level command without scrolling through the menu to find it.

NOTE: Version 5.00 of the Mouse Systems (IMSI) mouse driver has a bug causes the cursor movement in the Y-axis to be opposite of what is expexted. Prior versions and subsequent revisions do not have this problem. If you have version 5.00 of the Mouse Systems driver, contact Mouse Systems for an update.

3.11. WORKSHEET FILE STRUCTURE

Design organization is an integral part of the schematic design process. Typically, a design moves through the refinement of general concepts to a final set of detalied schematic diagrams. With OrCAD/SDT schematic worksheet may be created using three different file structure, depending on complexity. The file structure are:

* Flat File;

* Hierarchy File;

* One-Sheet.

Typically, you will select the file structure for creating your schematic worksheet that best suits the complexity of your design. We recommended that you become familiar with all three file structures. The file structure you select determines how you use module ports (graphic objects used to represent signal connections between worksheets) and labels in the worksheet, and furthermore, how you invoke OrCAD/SDT's utility programs.

Further information on using each file structure is found in Section 5 (Hierarchy), Section 6 (NETLIST Utility), and Section 8 (Tutorial).

3.11.1. The Flat File Structure

The flat file structure, traditional method for organizing schematic sheets, can be used for simple or complex design. This file structure can be through of as a collection of single worksheets with signals connected one worksheet to another through module ports (graphic objects used for connections between worksheets).

Internally, the worksheet contains parts, wires, buses, labels, junctions, and other objects. You simply create a fat file structure by assigning a separate filename to each schematic worksheet in your design, then connect inter-sheet signal through the module ports.

An example of a typical flat file structure is illustrated in Figure 3-4 below.

(No Figure !)

Figure 3-4. Flat File Structure

In this example, you see three separate worksheets with unique filenames: CPU.SCH, I/O.SCH, and MEMORY.SCH.

Signals (wires and buses) are connected one worksheet to another through identically named module ports, A(0..7) from the CPU.SCH worksheet is connected only to A(0..7) on the I/O.SCH worksheet. Likewise, the XYZ and B(0..15) module port connect the CPU.SCH and MEMORY.SCH worksheet together. Finally, C(0..3) connects the I/O.SCH and MEMORY.SCH worksheets.

For more information on module ports, see the PLACE module port command in Section 4.

3.11.2 The Hierarchy File Structure

While a flat file structure is easy to understand and use, keeping track of module ports and managing utility programs can be difficult when your design is large or complex.

Creating your worksheets in a hierarchical fashion is an easier way to organize a complex design. A hierarchical design has its individual worksheets organized in blocks, where each block contains part of the design. A block may have a block within it, containing progressively more detail. Hierarchical

blocks may be used over and over to complicate circuit functions.

The flat file structure of Figure 3-4 is shown modified into a hierarchical file structure in Figure 3-5 below.

(No Figure !)

Figure 3-5. Hierarchy File Structure; COMPUTER.SCH

In a hierarchical file structure, there is one "root" worksheet. In this example the root worksheet has the filename COMPUTER.SCH. Using the PLACE Sheet command, you place individual blocks (named CPU.I/O, and MEMORY) in the root worksheet. These blocks are called "sheet symbols", and they represent separate unique worksheets.

To display each sheet symbol as a worksheet, you place the cursor inside the sheet symbol boundary and execute QUIT Enter Sheet. This moves you one level down in the hierarchy.

If you enter the MEMORY sheet symbol for example, the worksheet represented by the MEMORY sheet symbol appears on the screen. With the OrCAD/SDT, you can create a hierachical structure of more than 5000 levels.

In the figure, connections between sheet symbols are made through "nets" placed along the borders of the sheet symbol. In a hierarchy, nets connect module ports to sheet symbols.

The nets, named XYZ, A(0..7), B(0..15), and C(0..3) in Figure 3-5, connect the CPU.I/O, and MEMORY sheet symbols. As in the flat file structure, module ports are used to label signals that leave a worksheet.

For more information on hierarchical file structure, see Section 5.

3.11.3. The One-Sheet File Structure

If the names implies, a one-sheet file structure is a design that is contained within one worksheet. Even through they are allowed, a one-sheet file structure design uses no module ports: all signals remain within the worksheet. One sheet file structure may be handled as either flat or hierarchy file structure when you invoke the post-processing utility programs (see Section 6).
.PA

4. COMMANDS

This section describes the OrCAD/SDT commands in the order they appear on the main command menu. Certain concepts, described below, are used repeatedly in different commands. See Section 3.1 for a review of command execution procedures.

Cursor Movement

You may use the cursor control keys or a mouse to move the

cursor on the screen.

Defining an Area

For many OrCAD/SDT commands to work, you must define an area on yhe worksheet where you want these commands to have an effect. Commands that ask you to define a worksheet area are: BLOCK Move, BLOCK Drag, BLOCK Save, BLOCK Export, DELETE Block, and PLACE Sheet. When you select one of these commands, DRAFT prompts:

Begin Find Jump Zoom escape

You define an area by drawing a box. Place the cursor where you want the comer of the box to begin (point A in Figure 4-1 below).

(No Figure !)

Figure 4-1. Placing the Cursor at a Begin Point

Select Begin. On the prompt line, DRAFT returns:

End Find Jump Zoom escape

As you move the cursor, a box is drawn that surrounds an area on the worksheet (see Figure 4-2 below).
.pn42

(No Figure !)

Figure 4-2 Drawing the Box

When you complete the box, select End. The area is now defined, as Figure 4-3 shows. Once the area is defined, you may proceed with the command that was originally selected.

(No Figure !)

Figure 4-3. The Cursor Placed at the End Point

NOTE:
The shape that defines an area may be a point, horizontal line, vertical line, or rectangle.

Locating Objects

To locate an object or a specific area on a worksheet, use these subcommands:

Find: To search for a string of characters. See Section 4.6.

Jump: To quickly move to a specific location. See Section 4.9.

Zoom: To change the worksheet scale. See Section 4.17.

Other concepts are:

Objects: Library parts, wires, buses, junctions, labels, sheet symbols, module ports, power objects, etc.

<RUBOUT> is the backspace key, and erases text.

Press <ESCAPE> to abandon a command or subcommand. This command returns you to the main command or previous subcommand level.

.he Commands AGAIN

4.1. AGAIN

Again repeats the previously executed main level command. For example, if you previously selected PLACE, you may repest PLACE with the AGAIN command.
.pa
.he Commands BLOCK

4.2. BLOCK

BLOCK and its subcommands enable you to manipulate specific areas of your worksheet. With the BLOCK subcommands Import and Export, you may also import previously saved files into your worksheet or write an area of your worksheet to a file.

Select BLOCK to move, rubberband, make orthogonal, replicate, import, or export a section of a worksheet. DRAFT returns this subcommand menu:

* Move
* Drag
* Fixup
* Save
* Get
* Import
* Export

4.2.1. BLOCK Move

To move an object or area of the worksheet, first define the area that you want to move: select Begin, surround the area, then select End.

When the area is defined DRAFT returns:

Place Find Jump Zoom Escape

Objects enclosed and intersected by the surrounding area may now bw moved. The objects are then converted to autlined symbol shapes that enable them to move quickly around the screen. If you select an area containing many objects, only the box that defines the area appears to move.

Move the area you have selected to the desired location.

Note that objects within the area being moved still remain at their original location. Only the outline symbols within the selected area move on the screen.

To place the moved objects in the new worksheet location, select Place. The screen is redrawn, placing the objects.

NOTE

You may move and place a single object by positioning the cursor inside the object, then subsequently selecting Begin and End. It is not necessary to enclose the object in a box. Move and place the object as described above.

.pa

4.2.2. BLOCK Drag

BLOCK Drag moves objects while maintaining connectivity for rubberbanding wires and buses.

To drag a bus, you must enable the SET Drag Buses option (described in Section 4.15.3.) to maintain bus connectivity.

BLOCK Drag executes the sames as BLOCK Move.

4.2.3. BLOCK Fixup

BLOCK Fixup enables you to "fix up" wires an buses, making them orthogonal by adding new segments. When you select Fixup, DRAFT returns this subcommand menu:

Pick Find Jump Zoom escape

To select Fixup, place the cursor on either end of (touching) the wire, bus, or node that you want to make orthogonal. Select Pick to attach the cursor. DRAFT returns this subcommand menu:

Drop End Find Jump Zoom escape

DRAFT displays a new conductor segment at the tip of the cursor when you move the cursor controls or mouse. Select Drop to attach the new conductor segment where you want it.

Select End when you have finished Fixup. DRAFT returns you to the "Pick Find Jump Zoom escape" menu, enabling you fixup another wire or bus.

NOTES:

1. If a node has more than one wire or bus connected, a menu enables you to select either Drag All or Pick One wire or bus.

   Drag All enables you to drag all the wires or buses attached to a common node.

   Pick One enables you to choose one wire or bus for fixup from those connected to a common node. When

you select Pick One, DRAFT returns you this subcommand menu:

Next
Previous
This

The wire or bus you select with Pick One is displayed in phantom (dashed lines). next selects the next wire or bus for fixup.

Previous selects the previous wire or bus for fixup. Select This to fix up the wire or bus that you have selected (dashed lines).

2. Use Fixup for straightening non-orthogonal wires and buses by adding segments to them. Clean-up application that do not need segments added should be done with BLOCK Drag.

4.2.4. BLOCK Save

BLOCK Save stores a group of objects for replications in another area of worksheet. Use the procedure in Section 4.0 to define the area you want to save.

When you select End after defining a worksheet area, the objects within the area you have defined to save in memory and DRAFT returns to the main command level. Objects that you have saved may be recalled to the screen with the BLOCK Get subcommand.

NOTE

The buffer used to save objects is also used by BLOCK Move and BLOCK Drag. Objects saved with a BLOCK Save will be lost after a following BLOCK Move or a BLOCK Drag is executed.

If you want to save objects and still use BLOCK Move or BLOCK Drag in your editing sesion, use BLOCK Export (see the BLOCK Export explanation) rather than BLOCK Save to store worksheet objects.

4.2.5. BLOCK Get

BLOCK Get retrieves objects that were saved via BLOCK Save. When you select BLOCK Get to retrieve objects stored in memory, DRAFT returns:

Place Find Jump Zoom escape

You will see a box containing the previously saved area with the cursor attached to its end.

Move the box to the desired worksheet location. Place the objects worksheet with the Place subcommand.

You may continue to place saved objects with BLOCK Get, or press <ESCAPE> to return to the main command menu.

.pa

4.2.5. BLOCK Import

With BLOCK Import you may retrieve stored in other files and place them in your current worksheet. To retrieve the contents of another file, select BLOCK Import, DRAFT returns:

File to Import?

Type the path and the filename of the file you want to import and press <ENTER>, DRAFT returns:

Place Find Jump Zoom escape

Pozition the cursor on the worksheet where you ant to place the contents of a file. Select Place to place the contents of the imported file on the worksheet. The imported objects are placed on the worksheet with the cursor in the upper left corner of the imported area.

4.2.6. BLOCK Export

With BLOCK Export you can save a defined worksheet area to a file. To export an object or worksheet area, define the area that you want to export (see 4.0 Defining an Area).

After you defined the worksheet area, select BLOCK Export, DRAFT returns:

Export file name?

Type the path and filename for the worksheet area you want to export and press <ENTER>. Objects enclosed and intersected by the outlined area are saved to the file you specified.
.pa

4.3 CONDITIONS

Conditions enables you to monitor your personal computer memory and the memory available for these items:

Worksheet

Hierarchy buffer

Macro buffer

Conditions has no subcommands. You may use the highlighted bar in the status menu as an aid in reading the table. To return to the main command level press either <ESCAPE> or <ENTER>.

What you see in the pop-up Conditions is described below.

4.3.1. Worksheet Memory Size

Shows worksheet memory size, in bytes, when Conditions was

invoked. Tells you how much memory your worksheet uses, and about how much disk space will be neded to save it.

NOTE
A worksheet uses memory space, even if it is blank, to hold frame and the title block information.

4.3.2. Free Hierarchy Buffer

Shows how much memory id available in the hierarchy buffer. The hierarchy buffer keeps track of sheet names when you move trough a worksheet hierarchy.

4.3.3. Free Macro Buffer

Shows the memory available in the macro buffer. Macros created on-line or loaded into the system from a macro file are stored in the macro buffer. If user-created macros become too large, the macro buffer is unable to store them. See Section 2.4.11 to increase the size of the macro buffer. See Section 4.11 for a description of macros.

4.3.4. Free System Memory

Shows how much memory remains in your Personal Computer.

.pa
.he Commands DELETE

4.4. DELETE

DELETE enables you to delete objects or blocks of objects. When you invoke DELETE, DRAFT returns a menu where you can select one of these subcommands:

* Object
* Block
* Undo

4.4.1. DELETE Object

As the name implies, you use DELETE Object to delete an object from the worksheet.

Select DELETE Object. On the prompt line, DRAFT returns:

Delete Find Jump Zoom escape

Place the cursor on the object you want to delete (you may use Find, Jump, and Zoom to help locate objects on the worksheet) and select the Delete subcommand.

If you want to delete one of two intersecting wires and you have placed the cursor at their intersection, the first wire drawn will be the first deleted. To delete the last wire drawn, move the cursor away from the untersection along the wire you

wnat to delete and delete it.

If the cursor is pointing to more than one object, DRAFT returns:

Delete wich Object?

DRAFT displays a meny listing objects to delete. Select the object from the menu to delete it.

NOTE

You must place the cursor within the body of the part to delete it.

When you have deleted an item from the worksheet, DRAFT returns you to the Delete subcommand level, where you may continue to delete objects.

To return to the main command level and redraw the worksheet, press <ESCAPE>.

4.4.2. DELETE Block

Use DELETE Block to delete a worksheet area. To delete an area of the worksheet, define the area (refer to Section 4.0 to Define an Area).

When you finish defining the area, objects within and intersected by the outlined area are deleted when you execute the End subcommand. Once the block of objects are deleted, DRAFT returns you to the main command level.

4.4.3. DELETE Undo

You may restore accidentally deleted objects to the worksheet with DELETE Undo. This command restores objects deleted with the last execution of the Felete command.

4.5. EDIT

Edit enables you to:

1. Edit the title block, module ports, labels, power objects, sheet symbols, part reference designators, and part names.

2. Select different pinouts on devices with multiple parts per package.

3. Move a part reference designator and name to other locations.

When you invoke EDIT, DRAFT returns:

Edit Find Jump Zoom escape

Follow the descriptions below to edit objects on the worksheet.

4.5.1. Editing Labels

Place the cursor under the label name and select EDIT. DRAFT returns:

Name Type Orientation

Select Name to edit the name of a label. When you select Name, the label name appears on the prompt line. Use <RUBOUT> (the Backspace key) to erase it. Type rhe new name on the prompt line, then press <ENTER> to place it in the worksheet.

Select Type to amke the label an internal, bus member, or comment label:

* Select Internal to change the label to an internal label

* Select Bus Member, to change the label to a bus member label

* Select Comment, to change the label to a comment label

For more information on labels, see 4.12.5 - 4.12.6, 4.5.2 Editing Module Ports.

To edit a module port, place the cursor within the module port symbol and select Edit. A menu displays these subcommands:

Name Type

Select Name to edit a module port name. When Name is invoked, the module port name appears on the prompt line after "Module Port Line?".

Use <RUBOUT> to erase the name. Type the new name and press <ENTER> to place it in the worksheet.

Select Type to change the type of module port. Module port choices are: input, output, bidirectional, or unspecified.
.pn51
.he Commands EDIT

Use <ESCAPE> to abandon editing a module port. DRAFT returns to the Edit subcommand prompt.

For information on placing module port on the worksheet, see 4.12.7. Place Module Port.

4.5.3. Editing Power Objects

To edit a power object, place the cursor on the power object, and invoke Edit. DRAFT returns:

Name Type Orientation

Select Name to edit the name of the power object. When you invoke Name, the name of power object appears on the prompt line after "Power Name?".

Use <RUBOUT> to erase it. Type the new name on the prompt line and press <ENTER> to place it in the worksheet.

Select Type to change the type of power object. The choises are: cicle, arrow, bar, or wavw.

Select Orientation to change the orientation to top, bottom, left, or right.

For information on placing power pbjects on the worksheet, see 4.12.7. PLACE Power.

4.5.4. Editing Sheets

To edit sheet symbols, place the cursor within the sheet symbol boundary and select Edit (use Find, Jump, or Zoom to locate the sheets you want: to edit).

When you invoke Edit, cursor movement is restricted to the border of the shee: symbol. This helps you place the cursor at sheet net name locations. When Edit is invoked, DRAFT returns:

Add Delete Edit Name Filename Size Zoom escape

Use Add to add net connections between worksheets. To add a net name, place the cursor at the edge of the box where you want to place the net name, then select Add.

DRAFT returns:

Net Name?

Type the desired net name and press <ENTER>. A menu displays input output, bidirectional, and unspecified net names. Select the kind of net name you want and press <ENTER> to place it on the worksheet.

To delete a net name, place the cursor on the net name and select Delete.

To edit a net name, place the cursor on the net name and select Edit. When the net name appears on the prompt line, press <RUBOUT> to erase it.

Type the new net name and press <ENTER> to enter it on the worksheet.

Name is used to edit the name of a sheet symbol, located on the top of the sheet. The initial sheet name is a question mark (?). Typically, names such as "Memory Array", or Dynamic RAM Refresh Circuitry", are used to identify the function of the worksheet represented by the sheet symbol.

To edit the name of the sheet symbol, select Name. Draft returns "Sheet name?" and shows the name on the prompt line. Press <RUBOUT> to erase the old name. Type the new name on the prompt line and press <ENTER> to place it at the top of the sheet.

Press <ESCAPE> if you want to abandon any changes made to the sheet name. Filename enables you to edit the filename that contains a worksheet by a sheet symbol.

DRAFT automatically generates a filename based on the date and time of day, ensuring that no two filenames will be alike. The filename generated by DRAFT appears on the prompt line when you invoke Filename.

If you want to edit the filename, select Filename. DRAFT returns "File name?" and shows the filename on the prompt line. Press <RUBOUT> to erase the olf file name. Type a new file name and press <ENTER> to enter it on the worksheet.

Press <ESCAPE> to abandon any changes made to the file name.

Select Size to change the size of the worksheet displayed on the screen. DRAFT returns:

End Jump Zoom escape

DRAFT places the cursor on the lower right corner of the worksheet. To change worksheet size, move the cursor until you reach the desired size, then select End.

For more information on worksheet sizes, see 4.12.8 Place Sheet, and Section 5. Hierarchy.

4.5.5. Editing Parts

Editing Parts enables you to edit and move part reference designators and values. It also enables you to select other packages on library parts with multiple parts per package and change the orientation of the symbol. Figure 4-4 ilustrates a library part with its default reference designator and part value.

(No Figure !)

Figure 4-4. A CMOS 4013 Library Part with ist Defaults Reference Designator and Part Value

To Edit a Part, place the cursor within the part symbol boundary and select Edit. For a description of a boundary see 4.7.1. The Outline Symbol.

When you select Edit, DRAFT returns a subcommand menu with the selections Reference, Part Value, and Orientation. "Which Device?" appears when you edit a device with multiple parts per package. The part editing subcommands are described below.

Reference

Select Reference to edit or move the reference designator values of library parts placed on the worksheet. Typical examples of reference designators are: U1, U2A, Q6, R1, R2, and C12.

To edit or move referece designators, select the Reference sheet command. DRAFT returns:

Name Location

Select name to edit the name of a reference designator. If the part has a reference designator naem assigned, it appears on

the prompt line. Press <RUBOUT> to erase it. Type the new reference designator name on the prompt line and press <ENTER> to place it on the worksheet.

Select Location to change the reference designator's location. DRAFT highlights the reference and part value designators and returns:

Place Find Jump Zoom escape

You may move the reference designator anywhere in the worksheet using the cursor keys or mouse. Place to place the reference designator in the new worksheet location.

Refer to the utility program Annotate (discussed in Section 6) for a method of automatically incrementing reference designators and changing the corresponding pin numbers of parts placed on the worksheet.

Part Value

Select Part Value to edit or move the part values of components on the worksheet. Typical examples of part values are: 100K, IN4004, .01, 2N2222, and 80386.

To edit or move part value names, select Part Value. DRAFT returns:

Name Location

Select Name to edit the part value. If the part has a part value assigned, it appears on the prompt line. Press <RUBOUT> to erase it. Type the new part value on the prompt line, then press <ENTER> to place it in the workshee.

Select Location to move the part value, DRAFT highlights the reference and part value designators and returns:

Place Find Jump Zoom escape

You may move the part value anywhere in the worksheet using the cursor keys ore mouse. Select Place to place the part value in the new worksheet location.

NOTE

If the token REFERENCE is used in library source files, the library part name is inserted in place of the part value when parts are placed in the worksheet. This may be edited to enter the actual part value.

Orientation

If you want to repositon a part, select Orientation. DRAFT returns:

Rotate Convert Normal Up Pver Down Mirror Zoom escape

Rotate

This subcommand rotates the part 90 degrees counterlockwise from its current position.

Convert

This subcommand is displayed when editing a part having another representation of the same part. For example, you may change a 74LS02, normally represented as a NOR gate, to a DeMorgan representation with Convert.

Normal

This subcommand returns a rotated part to its original position, as created in the part library. Normal also returns parts to a normal pozition that have been converted to other shapes with the Convert and Mirror subcommands.

Up

Up rotates a part once, 90 degrees counterclockwise, from its normal position.

Over

This subcommand rotates a part 180 degrees counterclockwise, equivalent to rotating it twice from its normal position.

Down

This subcommand rotates a part 270 degrees counterclockwise, equivalent to rotating it three times from its normal position.

Mirror

This subcommand gives a mirror-image of a part. Mirroring is along the horizontal axis.

Which Device

This subcommand only appears when you are editing a part that contains more than one part per package. An example would be a 74LS04 hex inverter. This part contains six inverters. You may select a different part in the package by invoking the Which Device subcommand. Select the appropriate number that represents the device in the package you desire.

4.5.6. Editing the Title Block

To edit title block information, place the cursor inside the title block and invoke Edit. The title block is in the lower right worksheet corner. The title block information that you add or edit goes to a holding buffer, appearing in the title block on screen after you press <ECSAPE> to end the title block edit.

When you invoke Edit, DRAFT returns a menu of choices described below:

Revision Code

Select this subcommand to add or edit the revision code. DRAFT returns, "Revision Code?" on the prompt line.

Type the desired revision number (three characters maximum). If you are editing the revision code press <RUBOUT> to erase, then type the new revision code information on the prompt line. Press <ENTER> to place it in the holding buffer. When you have finished editing the revision code, press <ESCAPE> to update the title block.

Title of Sheet

To add or edit a title (up to 44 characters maximum), select Title of Sheet. DRAFT returns, "Title of Sheet?".

Type the desired sheet title. If you are editing the worksheet title press <RUBOUT> to erase, then type the new title information on the prompt line. Press <ENTER> to place it in the holding buffer.

When you have finished editing the sheet title, press <ESCAPE> to update the titleblock.

Document Number

To edit or add a document number (up to 36 characters), select Document Number. DRAFT returns, "Document Number?".

Type the document number. If you are editing the document number press <RUBOUT> to erase, then type the new document number on the prompt line. Press <ENTER> to place it in the holding buffer.

When you have finished editing the document number, press <ESCAPE> to update the titleblock.

Sheet Number

To add or edit a sheet number (any number up to 32767), select Sheet Number. DRAFT returns "Sheet Number?".

Type the sheet number. If you are editing the sheet number press <RUBOUT> to erase, then type a new sheet number on the prompt line. Press <ENTER> to place the new sheet number information in the holding buffer.

When you have finished editing the sheet number, press <ESCAPE> to update the titleblock.

Number of Sheet

To add to the number of worksheet (any number up to 32767), select Number of sheets. DRAFT returns, "Number of Sheets?".

Type the desired number of sheets. If you are editing, press <RUBOUT> to erase, then type the new number of sheets on the prompt line.

Press <ENTER> to place the number of worksheet information in the holding buffer.

When you have finished editing the number of worksheet, press <ESCAPE> to update the titleblock.

Organization Name

To add or edit an organization name (up to 44 characters), select Organization Name. DRAFT returns: "Organization Name?".

Type the desired name. If you are editing, press <RUBOUT> to erase, then type the new organization name on the prompt line. Press <ENTER> to place it in the holding buffer.

When you have finished editing the organization name, press <ESCAPE> to update the titleblock.

Address Lines

With this command, you can edit any of the four address lines (up to 44 characters).

To add or edit on address line, select the desired address line (1 through 4) from the subcommand menu. DRAFT returns "Address Line?".

Type the desired address. If you are editing, press <RUBOUT> to erase, then type the new address on the prompt line. Press <ENTER> to place it in the holding buffer.

When you have finished editing the address, press <ECSAPE> to update the title block.

Size

Select the worksheet size with the Set subcommand. Refer to Section 4.15 for information on Set.

Date

When you create a worksheet, the date is automatically placed in the title block, and is updated when you make changes.

Revision History

You can add a revision history to the top of the titleblock with the PLACE/Wire and PLACE?Label/Comment commands.

Draw a revision history box with the PLACE/Wire command, and add next with the PLACE/Label/Comment command.

.PN58

4.6 FIND

Find locates a string of characters anywhere in a schematic worksheet and places the cursor at the object containing the search string. A search string can be any number of characters that are grouped within the following items:

Module ports
Labels

Reference designators
Part values
Sheet symbol names
Power objects

Find works only in the worksheet you are editing. When you select Find, DRAFT returns "Find?"

Type the character string that you want to find, then press <ENTER> DRAFT searches the worksheet for he desired character string and places the cursor near it.

The next time you select Find, you will see the previous string on the prompt line. To search for a new string, use <RUBOUT> to erase the previous entry and type the new character string.

If you are searching for a string identical to the last string (for example, you want to find all 200 ohm resistors on the worksheet), press <ENTER> with the current string name on the prompt line.
DRAFT remembers the location of the previous string and searches for the next one.

If you select Find after fiding the last occurrence of the string. DRAFT "wraps" to the first occurrence of the string. If there is only one occurrence of the string, Find returns repeatedly to the string when the command is invoked.
.PA

4.7 GET

Get retrieves objects from the part library database and places them in the worksheet as normal, rotated, or converted symbols. There are two ways objects can be retrieved from the library database.

1. Select Get. DRAFT returns "Get?"

Type the desired object name exactly as it appears in the part library directory. If the name yped does not match the library directory, the prompt line shows an error message. To verify the spelling of an object name, use the LIBRARY Directory command.

Press <ENTER> to show the symbol outline on the screen.See Section 4.7.2 for information on rotating and placing parts on the worksheet.

2. Select Get. DRAFT returns "Get?"

Press <ENTER>, DRAFT retuns a subcommand menu displaying a list of part libraries. Select the library that you want to get a part from.

When you select a library, a menu shows the select library parts directory. Scroll the reverse video bar to the part name you select, then press <ENTER> to retrieve the part. The part outline symbol is placed on the screen.

See Section 4.7.2 for information on rotating and placing parts on the worksheet.

You can select TTL and other library part numbers created with a prefix and shorthand string (see Section 7, The Prefix Definition) from the library by entering the suffix. For example, suppose you want to retrieve a 74LS27 from the library. After invoking Get, you can use any of the fillowing entries to retrieve the part:

A. Get? 74LS27 <ENTER>

B. Get? LS27 <ENTER>

C. Get? 27 <ENTER>

Example A uses the entire part name to retrieve the part. Example B uses the LS portion on the prefix to retrieve the part. Example C uses only the sufix of the part name. Since the prefix was not defined in example C, DRAFT displays a menu of all available TTL parts using "27" as a suffix.

.PA

4.7.1 The Outline Symbol

After you get the part from the library, the screen shows an outline of the part symbol. The outline symbol shows the size and shape of the part but contains little detail. Its function is to enable the part to move quickly around the worksheet.

If the outline symbol remains stationary, the actual part symbol apears on the screen enclosed within the outline symbol. This enables you to view the layout of the part as it would appear when placed in the worksheet.

4.7.2 Rotating and Placing Parts

With the part selected and the outline symbol on the screen, DRAFT returns:

Place Rotate Normal Up Over Down Mirror Find...

...Jump Zoom escape Convert

Move the symbol to where you want to place it. You may use these subcommands to rotate or place the part in the worksheet.

Place

Select this subcommand to place the part in the worksheet.

Rotate

Selecting this subcommandrotates the part counterclockwise 90 degrees. Part rotate in the sequence up, over, down, and normal (Figure 4-5 shows a rotated part).

Normal

Use this subcommand to rotate a part to its original position, as retrieved from the library. This subcommand also returns parts that have been mirrored or converted to other shapes with Convert
to their original position (Figure 4-5 shows a part placed in its normal position).

Up

Select this subcommand to rotate a part 90 degrees counterclockwise (equivalent to rotating it once from its normal position). Figure 4-5 shows a part placed in the up position.

Over

Select this subcommand to rotate a part 180 degrees counterclockwise (equivalent to rotating it twice from its normal position). Figure 4-5 shows a part placed in the over position.

(No Figure !)

Down

Select this subcommand to rotate the part 270 degrees counterclockwise (equivalent to rotating it three times from its normal position). Figure 4-5 shows a part placed in the down position.

Mirror

Select this subcommand to get a mirror image of a part. Figure 4-5 shows a part placed in the mirror position.

Convert

Many library parts have DeMorgan or other equivalent symbol forms. If a part has an equivalent symbol form, the Convert subcommand appears on thr prompt line when it is retrieved from the library.

Select Convert to convert the part to its alternate symbol form. You may see the converted part by leaving the outline symbol stationary.

To return the converted part to its original symbol, select Normal.

After you place a part on the worksheet, DRAFT keeps the same part selected. This enable you to repetitively place it without repeating the selection process. When you have placed all the parts, press <ESCAPE> to return to the main command level.

NOTE

When an outline symbol is placed over the same part already placed on the worksheet, the part seems to disappear. Moving the outline symbol will display the originally placed part.
.PA

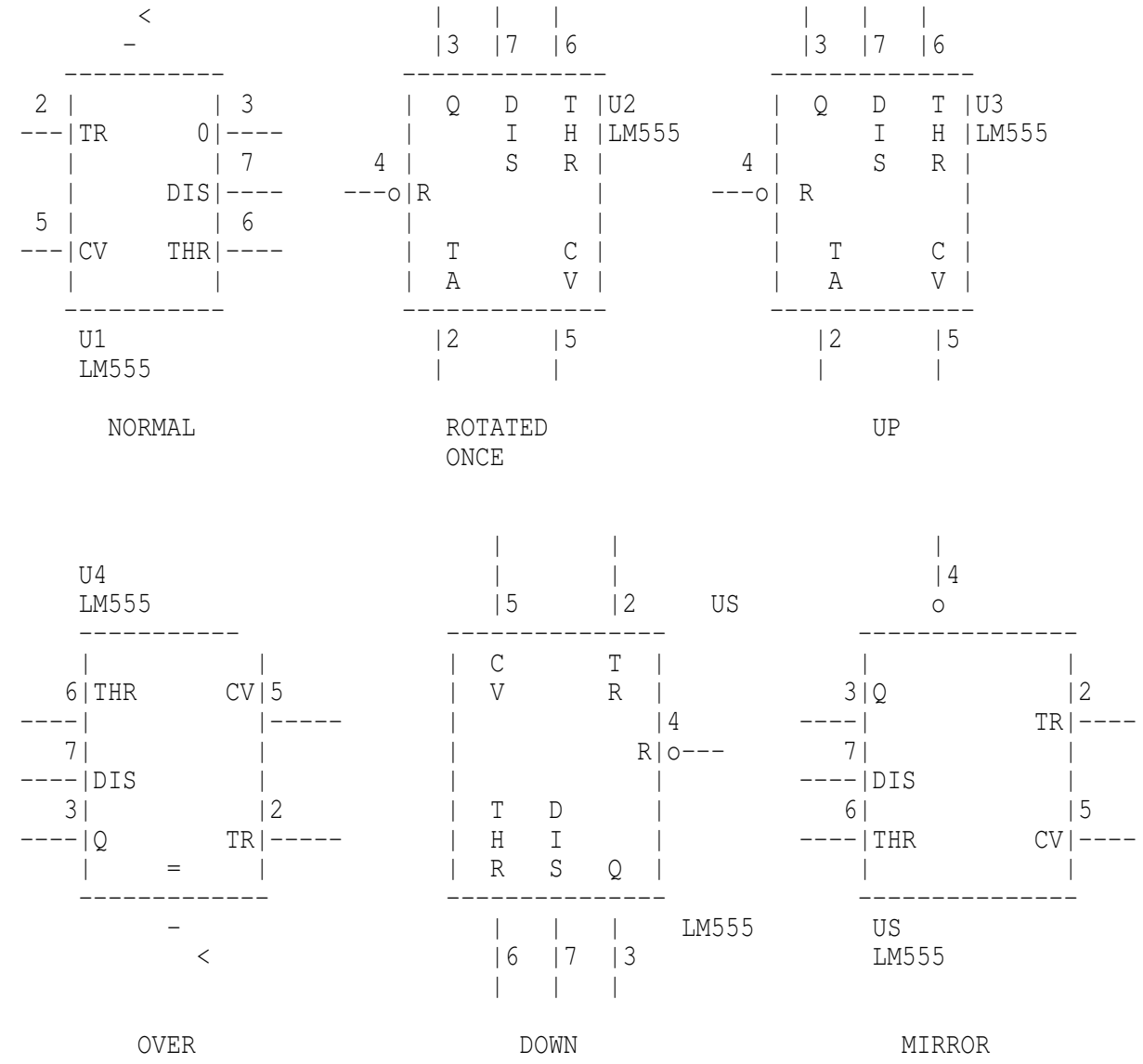


Figure 4-5. Parts Placed in the Normal, Rotated, Up, Over Down, and Mirrored Positions

.pa

4.8 HARDCOPY

The Hardcopy command enables you to send a worksheet to the printer or to a file while in DRAFT. TThe hardcopy command is not used to output to plotters. If you wish to plot a schematic, refer to the Plotall utility program in Section 6.

DRAFT can make printouts of all worksheet sizes, in both scale and compressed modes. Worksheets are printed in the normal printer orientation (horizontally). If a worksheet is too large to be oriented normally, it is printed sideways (vertically), and

if still larger, it is printed normally in several sections.

To make a printout, select Hardcopy from the main command menu. DRAFT returns this screen menu:

* Destination
* File Mode
* Make Hardcopy
* Print Mode
* Width of Paper

NOTE

Be sure that your printer is connected to parallel port 1. Other parallel ports are not supported.

4.8.1 HARDCOPY Destination

This subcommand selects the hardcopy destination. Select either LPT: or File from the menu.

Choose LPT: to send your worksheet to a printer.

Select File to send the worksheet to a binary file. Invoking File returns, "Destination of Hardcopy?"

Type the path and name of the file that you want the hardcopy seved to (be aware that since it is a graphics file, a lot of disk space is required). For example, entering:

B: FRED.PRN

sends the worksheet to a file called FRED.PRN on a floppy disk located in Drive B.

Thes may be sent to the printer using the DOS COPY Command. From the previous example, you would enter at the DOS prompt:

COPY FRED.PRNpm:/b

For more information on COPY, refer to your DOS User's Manual.

To return to the main command level, press <ESCAPE>.

4.8.2 HARDCOPY File Mode

File Mode enables you to append or replace the contents of the hardcopy file. Select either the Appended or Peplaced subcommands.

The Appended subcommand adds new data to the contents of the destination file. This subcommand enables you save a series of hardcopies to the same file name.

The Replaced subcommand replaces the contents of the destination

file with new data. This subcommand erases the contents of the destination file, then writes to the file.

Press <ENTER> to return to the Hardcopy command.

4.8.3 HARDCOPY Make Hardcopy

Select this subcommand to print a hardcopy of the worksheet that you have displayed on the screen. To make a Hardcopy of your worksheet, turn your printer on and, ensuring that it is on-line, select the Make Hardcopy subcommand. DRAFT returns:

:::Creating Hardcopy of Sheet:::

After a few seconds, the worksheet starts printing.

For printing hierarchy or flat file structures, use the PRINTALL utility program discussed in Section 6.

The Print Screen (PrtSc) key sends the worksheet contents that are displayed on the screen to a printer.

4.8.4 HARDCOPY Print Mode

Select the Compress or Scale subcommands from the menu. Compress prints the worksheet in compresed mode. The worksheet is printed in the highest density that the particular printer can handle.

The Scale subcommand prints the worksheet in the normal scale, 1:1, where one inch on the worksheet equals one inch on the printout.

Press <ESCAPE> to return to the Hardcopy command.

NOTE

The EPSON MX printer does not support output in the compressed mode. Output is always in Scale mode.

4.8.5 HARDCOPY Width of Paper

Select the 8 or 13 subcommands from the menu. 8 prints your worksheet on eight-inch wide paper, and 13 prints it on thirteen-inch wide paper.

Press <ESCAPE> to return to the Hardcopy command.

.pa

4.9 JUMP

Jump enables you to quickly move the cursor to specific location on the worksheet. The specific location can be tags, grid

references, or X-Y coordinates. For information on the tag command, see 4.16, the TAG Command.

When you select Jump, DRAFT returns a screen menu where you can select from these subcommands:

* Tag
* Reference
* X-Location
* Y-Location

4.9.1 JUMP Tag

When you select Tag, the cursor jumps to the specified tag on the worksheet (the tag must have been previously set with the tag command).

NOTE

The error message "Tag does not exist" is displayed if the specified Tag has not been set.

4.9.2 JUMP Reference

The Reference subcommand moves the cursor to a specified grid reference on the worksheet border. Grid references are invisible until you set them the SET Grid Parameters command. For information on grid parameters, see SET Grid Parameters, 4.15.12.

To jump to a grid reference, follow these steps.
.PN67
.HE Commands JUMP

1. Select the Reference subcommand.

2. DRAFT returns "Jump to Reference". Select the desired Y-axis alpha grid reference (A,B,C or D) from the menu.

3. Select the desired X-axis numeric grid reference (1 through 8) from the menu.

4. The cursor jumps to the grid reference location that you specified and you returns to the main command level.

4.9.3 JUMP X Location

This subcommand moves the cursor a specific distance in the X-direction. Each incremental step represents 1/10th (0.1) inch on the worksheet if the SET Grid References Stay On Grid command is enabled: otherwise it is 1/100th (0.01) inch. The procedure for X-Location jumps is outlined below.

1. Select the X-Location subcommand.

2. DRAFT returns "Jump X". Enter the number of steps you want to

jump. A positive number moves the cursor to the right, and a negative number (-10, -2.5, -30, etc.) to the left, the number of steps specified. If you enter 10 or +10, for example, the cursor jumps to the right 1 inch if you have enabled the SET Grid References Stay On Grid command.

3. When you press <ENTER>, the cursor jumps to the grid reference location that you specified and you returns to the main command level.

4.9.4 JUMP Y Location

This subcommand moves the cursor a specific distance in the Y-direction. Each incremental step represents 1/10th (0.1) inch on the worksheet if the SET Grid Reference Stay On Grid command is enabled: otherwise it is 1/100th (0.01) inch. The procedure for Y-Location jumps is outlined below.

1. Select the Y-Location subcommand.

2. DRAFT returns "Jump Y". A positive number moves the cursor down, a negative number (-10, -2.5, -30, etc.) up, the number of steps specified. If you enter 10 or +10, for example, the cursor jumps down 1 inch if you have selected t,he SET Grid Reference Stay On Grid command.

3. When you press <ENTER>, the cursor jumps to the grid reference location that you specified and you return to the main command level.

4.10 LIBRARY

The Library command enables you to display library part list directories and view the parts in libraries that are configured with DRAFT. When the library command is invoked, DRAFT returns a selection menu:

* Directory
* Browse

4.10.1 LIBRARY Directory

The directory subcommand enables you to select a library and output its parts directory to screen, printer, or a file. When[you select LIBRARY Directory, DRAFT returns a menu that displays a list of libraries that are currently configured in DRAFT.Select the library that you want a directory of from the menu.

The next screen menu enables you to choose the output device (Screen, Printer, or File).

Screen

This subcommand sends the library directory to screen. Press any key to continue.

Printer

Select this subcommand to send the library directory to a printer.

File

Select this subcommand to output and save the library directory to a file. DRAFT returns "File?" on the prompt line. Enter the path and filename and press <ENTER>..

4.10.2 LIBRARY Browse

The browse subcommand enables you to view the contents of a library, or select a part and view it on the screen. To browse through the part libraries select the browse subcommand. Select All Parts or Specific Patrs from the subcommand menu.

NOTE

Some devices may be too large to fit entirely on the screen. These devices may be viewed with the Get command.

All Parts

Select this option to view all parts in a library. DRAFT returns a menu showing a list oof the libraries that are currently configured. Select the library that you to view. A menu shows these subcommands: Forward, Backward, Quit.

Select Forward or Backward to browse through the library.

Quit returns you to the main command level.

Specific Parts

Select this option to view parts from the librarie. DRAFT returns "Part?".

.pa
.HE Commands MACRO

4.11 MACRO

The macro command enables you to capture, delete, initialize (erase), list, write to, and read macros from a file.

Schematic capture often involves repetitive tasks such as creating memory arrays, connecting wires and buses, or labeling items in the worksheet. Keystroke commands that are used to perform these tasks can be stored as macros, assigned to a key or key combination, then replayed by pressing the assigned key. This makes the entry of oftenused commands less tedious.

DRAFT can recorder over 100 key board macros. These macros can be assigned to: function keys,, selected keyboard keys, keys used with <CONTROL>, <SHIFT>, and <ALT>, or the middle button on a thee-button mouse.

When you create macros or load them from a macro file they are stored in memory allocated as a macro buffer. The macro buffer defaults to 16,384 bytes of memory. If the buffer fills, the prompt line shows a warning message. To get more buffer memory tou can either increase the buffer size (see Section 2.4.11) or delete unused macros.

To capture, delete, initialize, list, read, or write macros select the macro command from the main command level menu. DRAFT returns a selection menu of these subcommands:

* Capture
* Delete
* Initialize
* List
* Read
* Write

4.11.1 Valid Macro Key Names

Figure 4-6 shows a list of the key names that may be assigned as macros. The keys are illustrated as you would find them in a macro file. In the macro names, the following syntax has been used. The key names are as they appear on the keyboard with the exception of MMB which is the macro name of the Middle Mouse Button.

... is the name of the key that is pressed (such as HOME or A)

{...} the curly braces indicate that the name...is a macro.

^... indicates that the Ctrl key is pressed along with...

\... indicates that the ALT key is pressed along with...

SHIFT... indicates that the SHIFT key is pressed along with...
.PA

(^A)	(^B)	(^C)	(^D)	(^E)
(^F)	(^G)	(^I)	(^J)	(^K)
(^L)	(^N)	(^O)	(^P)	(^Q)
(^R)	(^S)	(^T)	(^U)	(^V)
(^W)	(^X)	(^Y)	(^Z)	
(^\)	(^])	(^^)	(^_)	
(\0)	(\1)	(\2)	(\3)	(\4)
(\5)	(\6)	(\7)	(\8)	(\9)
(\-)	(\=)			
(\A)	(\B)	(\C)	(\D)	(\E)
(\F)	(\G)	(\H)	(\I)	(\J)
(\K)	(\L)	(\M)	(\N)	(\O)
(\P)	(\Q)	(\R)	(\S)	(\T)
(\U)	(\V)	(\W)	(\X)	(\Y)
(\Z)				
(F1)	(F2)	(F3)	(F4)	(F5)
(F6)	(F7)	(F8)	(F9)	(F10)
(^F1)	(^F2)	(^F3)	(^F4)	(^F5)
(^F6)	(^F7)	(^F8)	(^F9)	(^F10)
(SHIFT-F1)	(SHIFT-F2)	(SHIFT-F3)	(SHIFT-F4)	(SHIFT-F5)
(SHIFT-F6)	(SHIFT-F7)	(SHIFT-F8)	(SHIFT-F9)	(SHIFT-F10)

(\F1)	(\F2)	(\F3)	(\F4)	(\F5)
(\F6)	(\F7)	(\F8)	(\F9)	(\F10)
(BACK TAB)	(DEL)	(INS)	(^RIGHT)	(^LEFT)
(END)	(HOME)	(PGDN)	(PGUP)	(^PGDN)
(^PGUP)	(D)	(L)	(R)	(U)
(MMB)	(MACROBREAK)			

Figure 4-6. Valid Macro Key Names

4.11.2 MACRO Capture

To create a macro, select the capture subcommand. DRAFT returns "Capture macro?".

Press the key or keys you want to use for the macro label. The key(s) that you pressed appears on the prompt line. Refer to Figure 4-6 above, for a list valid keys that can be assigned as macros.

Press <ENTER>. DRAFT returns "<macro>", informing you that you are in the macro capture mode.

In the macro capture mode, DRAFT records any sequence of keystrokes or mouse commands, commands that you normally perform in DRAFT can be recorder and executed later as macros.

When you finish recording the macro keystrokes, press <M>. You can execute the macro anytime by pressing the key that you assigned.

NOTE

A macro can execute another macro or call itself from withim a macro.

4.11.3 Macro Examples

1. This example macro displays the TTL parts directory to the screen. We'll assign the <F1> key to display the directory instead of sequencing through the commands.

a. Select the macro command from the main command menu.

b. Select the capture subcommand.

c. Press <F1> at the "Capture macro?" prompt (assigns F1 to the macro), then press <ENTER>. You are now in the capture macro mode.

d. Select the library command from the main command menu by pressing <L>.

e. Select the directory subcommand.

f. Select the TTL library.

g. Select the screen subcommand.

h. The directory is now displayed on the screen. Press <ENTER> twice to continue.

i. Press <M> to leave the macro capture mode.

To execute the macro press <F1>.

2. Using the PLACE command, this macro places junctions on the worksheet at the cursor location. We'll assign Ctrl A as the macro key.

a. Select the macro command from the main command menu.

b. Select the capture subcommand.

c. Press the <Ctrl A> keys at the "Capture macro?" prompt, then press <ENTER>. You are now in the capture macro mode.

d. Press <P> for the PLACE command.

e. Press <J> for the junction subcommand.

f. Press <P> to place the junction in the worksheet.

g. Press <ESCAPE> to return to the main command menu.

h. Press <M> to leave the macro capture mode.

To execute the macro press <Ctrl A>.

4.11.4 Terminating a Macro between Commands

You can capture and terminate a macro in the middle of a command sequence by simultaneously pressing <CTRL> and <END>.

For example:

We'll create a macro to simplify the procedure for drawing wires and assing it to F2. The macro will terminate after begin in the "PLACE Wire Begin" command.

Enter the following from the keyboard:

a. Press <M> for macro, then press <ENTER>.

b. Press <C> for capture macro, then press <ENTER>.

c. Press <F2>, then press <ENTER>.

d. Press <P>, <W>, and <B>, for "place, wire, and begin".

e. Press <CTRL> and <END> simultaneously.

When F2 is pressed, you are automatically placed into the "PLACE Wire Begin" mode. To draw a wire, mmove the cursor around the

worksheet with the cursor keys or mouse.

Capturing a Macro and Pausing for a Keyboard Entry

It is often handy to have a macro pause, enabling you to enter a keyboard command, before continuing. This is done you create the macro by pressing <CTRL> <HOME> simultaneously where you want the macro to pause for keyboard input.

Example:

Let's assign the macro we create to F3. We want to scroll through the Intel library directory, retrieve a part, move it to a worksheet location, place it, and returns to main command menu.

Enter the following:

a. Press <M> for macro, then press <ENTER>.

b. Press <C> for capture macro, then press <ENTER>.

c. Press <F3>, then press <ENTER>.

d. Press <G> GET, then press <ENTER>.

e. A window shows the libraries. Place the highlighted ber over the Intel library, then press <ENTER>.

f. Press <CTRL> <HOME> simultaneously. This causes the macro to pause and wait for you to scroll the window and select a library part. When the macro is invoked it pauses and remains at his step until you <ENTER>.

g. When you have selected a part, press <ENTER>.

h. Press <CTRL> <HOME> again simultaneously. The macro now pauses, enabling you to move the selected library part around the worksheet. You may place, rotate, etc. the part. The macro pauses and remains at his step until you press <ENTER>.

i. Press <ESCAPE> twice (this returns you to the main command level).

j. Press <M> to close the macro.

When you press F3, the macro runs and stops with the Intel library window showing on the screen. You select the desired part, then press <ENTER>. The macro continues, then stops, displaying the "Place, Rotate..." menu. You may move the part around the worksheet and place it by pressing <ENTER>. DRAFT then returns you to the main command menu.

NOTE

A macro that has paused is waiting for <ENTER> to be pressed. However, these is no indication that this is the case. Furthermore, if you try to execute a new macro and have not pressed <ENTER> on the current one, nothing happens. You must

press <ENTER> to finish the current macro.
.PN75

4.11.6 Executing a Macro

To execute a macro, press the key or key sequence assigned to it. For example, if the Alt F5 keys are assigned to a macro, press the two simultaneously.

4.11.7 MACRO Delete

To delete a macro select the delete subcommand. DRAFT returns "Delete macro?"

Enter the execution key (macro name) for the macro that you wish to delete, then press <ENTER>.

4.11.8 MACRO Initialize

This subcommand erases all macros. To erase all of the macros, select the Initialize subcommand. DRAFT returns "Erase All Macros?"

Select "No" to return to the main command menu, or "Yes" to erase all macros.

4.11.9 MACRO List

This subcommand shows a list of all the key names assigned to macros. To display the macro list select the list subcommand.

4.11.10 MACRO Read

The Read subcommand enables you to load a macro file into DRAFT. To load a macro file select the Read subcommand. DRAFT returns "Read all macros from?"

Enter the path and file name that the macros are stored in, then press <ENTER> to load the macro file.

4.11.11 MACRO Write

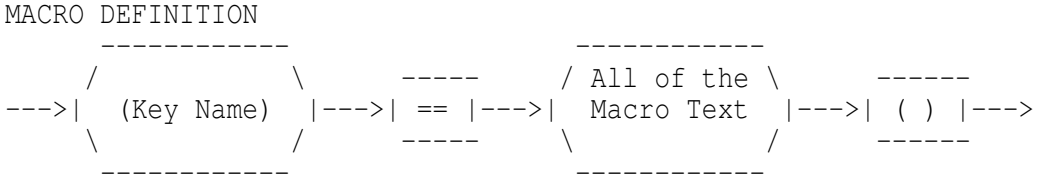
This subcommand enables you save all macros currently in DRAFT to a file. To save the macros to a file select Write. DRAFT returns
"Write all macros to?"
Type the path and filename and press <ENTER> to write the macros to the file.

File

Select.

4.11.12 Macro File Format

A macro format is a simple ASCII file, and may be edited or created using a text editor. If you want to examine or modify a macro, the syntax of a macro definition is shown in Figure 4-7 below:



Where: "Key Name"is a valid Macro Key.
"All of the Macro Text" are
the components of the Macro.

Figure 4-7. Macro Definition Syntax Diagram

From the left side, the macro has its key name enclosed in curly brackets (keyname). An equal sign (=) is next, showing that the macro text follows. Next is the macro text. Finally, the macro is terminated with a left curly bracket followed by a right curly bracket ({}).

Example:

Here is an example of a macro defined in a text file:

{F1}=g2115 {ENTER}p{ECS} {}

This macro used the function key F1 as a key name (F1 is enclosed in curly brackets). The macro text follows. The "g" executes GET, and 2115 says that library part 2115 is to be retrieved from the library. {ENTER} executes the <ENTER> command, p executes PLACE, and {ESC} executes the <ESCAPE> command. Finally, the curly brackets terminate the macro.

4.11.13 Nesting Macros

You may also nest macros, in which one macros calls another. To nest a macro, press the key name of a previously saved macro when you are capturing a new one. For example: If you are assigning F3 as a new macro, and you want to nest the macro assigned to F2 within the new macro, press F2 at the appropriate time while you are creating the new macro.

To nest a macro in a file, insert the macro key, enclosed by curly brackets, inside the text of another macro. For example:

{F3} = g2115 {F2} {}

4.12 PLACE

The place command enables you place wires, buses, junctions,bus entries, labels, moduleports, power, dashed lines, and hierarchical sheets on your worksheet.

Select place from the main command menu. DRAFT returns a selection menu with these subcommands:

* Wire
* Bus
* Junction
* Entry (Bus)
* Label
* Module Port
* Power
* Sheet

4.12.1 PLACE Wire

To place wires in the worksheet, select the wire subcommand. DRAFT returns:

Begin Find Jump Zoom escape

To draw a wire, place the cursor on the worksheet where you want the wire to stat. Follow the procedure outlined in the next paragraph.

Select Begin. DRAFT returns:

Begin End New Find Jump Zoom escape

Draw the wire by moving the curso. Select one of the following subcommands to finish drawing the wire:

Begin

Repetitively entering begin while the wire is beging drawn enables you to redefine the wire's origin where it makes a 90 degree turn.

To continue drawing the wire from the 90 degrees turn, select begin where the turn starts (point A in Figure 4-8). You may also move to the end of the wire (point B in Figure 4-8) and select either begin, end, or new, to fill it in. Dashes show a wire that has not been filled in by using the begin, end, or new subcommands.
.pa

Begin End Find Jump Zoom escape

|

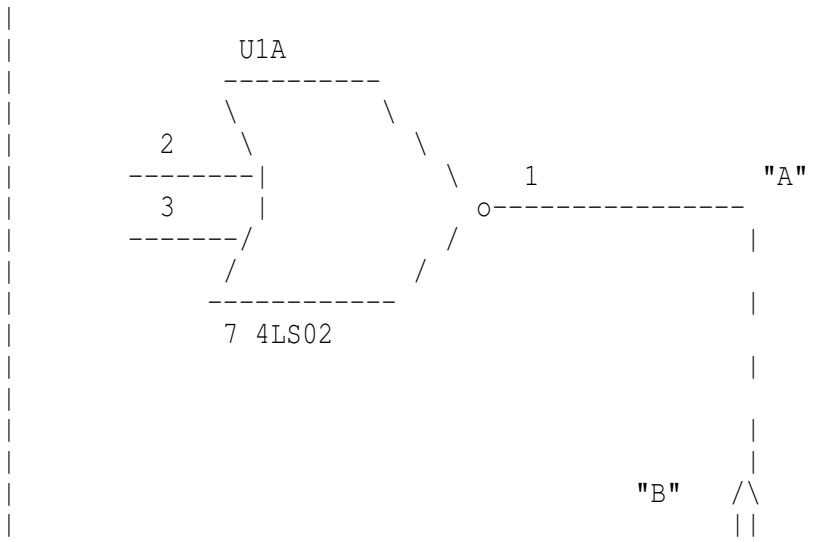


Figure 4-8. Drawing a Wire

Continue drawing the wire until you come to where you want it to end. To connect the wire to an end point, select either the end or new subcommands.

End

With the cursor at the end point select end. When you invoke this command, the program returns you to the main command menu.

New

With the cursor at the end point select new. This subcommand enables you to remain in the wire placing mode, and returns you to the Begin, Find, Jump, Zoom, and escape prompts.

NOTE

See Section 4.11.4 for example macros that simplify wire placement.

CAUTION

Wire and bus ends must not overlap library parts pins or each other. NETLIST and ERCHECK interpret these as "opens". Always place wire and bus ends end-to-end.

.pa

4.12.2 PLACE Bus

To place buses on the worksheet select the bus subcommand. DRAFT returns:

Begin Find Jump Zoom escape

To draw a bus, place the cursor at the worksheet location where you want the bus start. Select begin. DRAFT returns:

Begin End New Find Jump Zoom escape

Draw the bus by moving the cursor, then select end to complete it.

Drawing a bus is identical to drawing a wire (see 4.12.1, PLACE Wire command).

NOTE

If you are using the NETLIST utility program, you must label each with an internal label as follows:

BUSNAME [0..n]

Where "n" is tbhe decimal number of the last bus member, and BUSNAME is the name of the bus. See Section 6, NETLIST, for more information.

4.12.3 PLACE Junction

On a worksheet, many wires and buses connect or cross each other. Junction are pl5ced on the worksheet to distinguish a connection from a cross-over. If you inted to have more that two wires or buses connect to a common node, always place a junction at that point. This tell the ERCHECK and NETLIST utility programs that the node is a physical connection.

If you don't place a junction at an intersection of wires or buses, ERCHECK and NETLIST interpret the intersection as a cross-over.

A any design, you may want to connect a wire at 90 degrees to a bus. If you do, you must place a junction at the connect point. Junction are not required if you use a bus entry (see Section 4.12.4).

To place a junction in the worksheet, select the junction subcommand. DRAFT returns:

Place Find Jump Zoom escape

Position the cursor where you want the junction and follow the procedure for the place subcommand (described below).
.pa

Place

Use this subcommand to place the junction where you want it on the worksheet. The program remains in the "PLACE Junction" mode until <ESCAPE> is pressed.

4.12.4 PLACE Entry (Bus)

The entry (bus) subcommand enables you to place bus entries on the worksheet. Bus entries are used for aesthetic purposes to connect wires or other buses to a bus. Figure 4-9 shows wire bus

entires (A) and a bus entry used in a bus turn (B).

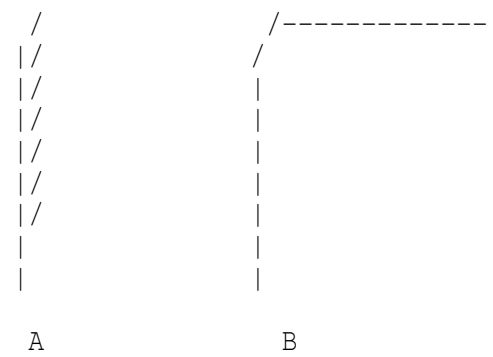


Figure 4-9. Illustrations of Bus Entries

When you select PLACE entry, DRAFT shows the last bus entry selecte, with this subcommand manu:

Place / \ Wire Bus Find Jump Zoom escape

To place a bus entry, select the place subcommand.

Select the / or\ subcommands to change the bus entry angle. DRAFT shows the last bus entry angle when you invoke the place entry (bus) command.

Select the wire subcommand to place wire thickness entries. Use this subcommand when a wire is to exit or enter a bus from another object.

Select the bus subcommand to place bus thickness entries. Use this subcommand when a bus makes a turn or is joined to another bus.

NOTE

Junction are not required to be placed in the worksheet to connect an angled bus entry to a bus.

4.12.5 PLACE Label
.pa
A label is an identifier placed on a worksheet to connect signals (wires and buses) together without actually physicall connecting them. You can place three kinds of labels horizontally or vertically on a worksheet: internal, bus member, or comment.

Internal or bus member labels are placed on wires and buses, and comment labels may be placed anywhere on the worksheet.

The place a label, select the label subcommand.DRAFT returns: "Label?"

Type the name of the label, then press <ENTER>. DRAFT returns this subcommand menu:

Internal

Bus Member
Comment

Select internal if the label identifies a bus or connect signals together. Select bus member if the label identifies a member of a bus. See the NETLIST utility program in Section 6 for detailed information.

Select comment for a comment label. Comment labels useful for placing revision history, tolerance, and other information in the worksheet.

NOTE

If you use the NETLIST utility program, you must follow the procedure for internal and bus member labels. Refer to the NETLIST utility program in Section 6 for detailed information.

When you select the kind of label you want, label name appears on the screen and may be moved anywhere on the worksheet before final placement. DRAFT shows this subcommand menu:

Place Orientation Value Type Find Jump Zoom escape

Select Place to place the label. When the label is placed, you are returned to the "Label?" prompt, where you may press <ESCAPE> to return to the main subcommand menu.

Select Orientation to change the label placement to horizontal or vertical. Selecting Horizontal or Vertical change the label placement accordingly.

Select Value to return to the "Label?" prompt. From here you can modify the last label that you entered.

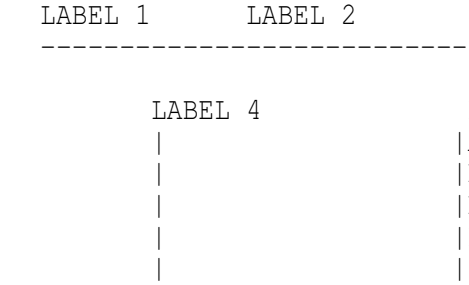
Select Type to change the label type to either internal, bus member, or comment.

.pa

4.12.6 Correct Label Positioning

For the ERCHECK and NETLIST utility programs to associate internal and bus member labels to wires and buses, labels must be placed with the leftmost point of the label name next to the bus or wire. The bottom of the leftmost character is the "hotpoint"; it must have contact with the bus or wire.

Figure 4-10 shows correct label positions for both vertical and horizontal wires. Notice LABEL4, the hotpoint (lower portion of the character "L") is closest to the wire.



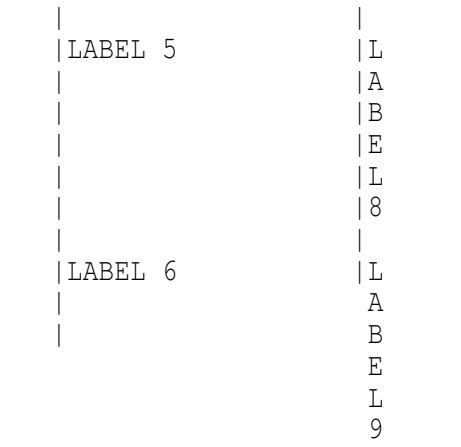


Figure 4-10. Correct Label Position

Figure 4-11 shows incorrect label position. None of label hotpoints are next to the wire. LABEL21 for example, has its hotppoint (lower portion of the character "L") away from the wire.
.pa

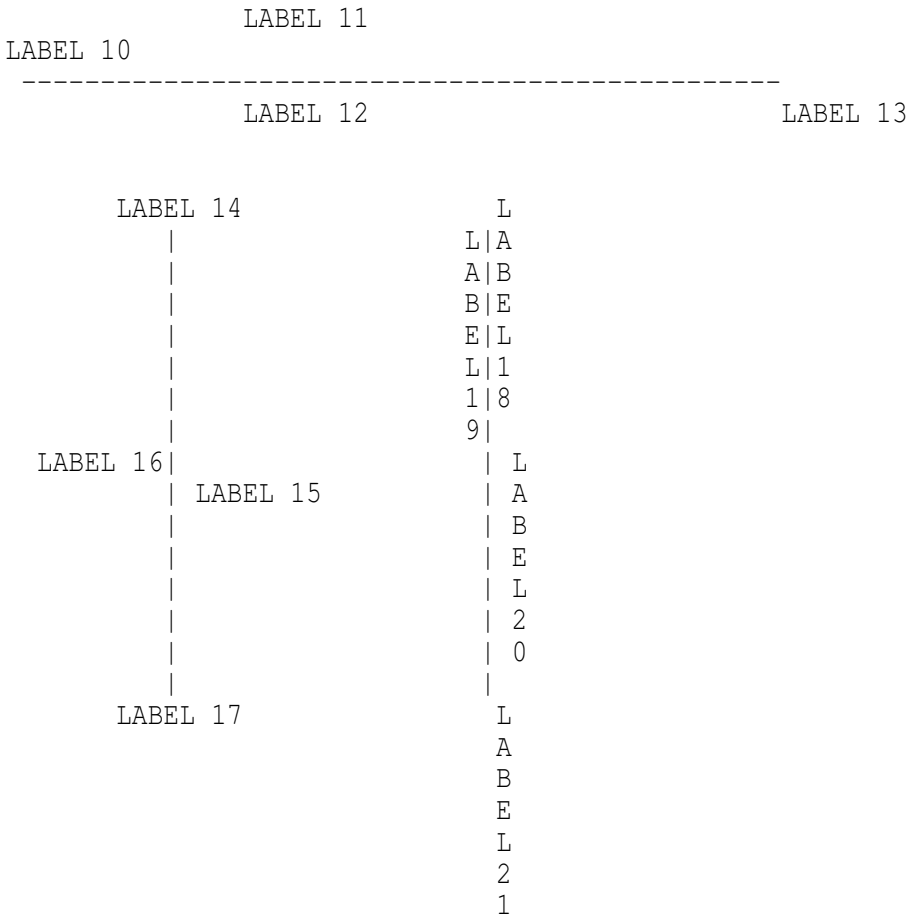


Figure 4-11. Incorrect Label Positions

4.12.7 PLACE Module Port

A module port connects hirarchical and flat file signals that leave a worksheet to signals in other worksheet having the same name. Unspecified module ports are also used to transfer power from one sheet to another. Module ports may be connected to

either wires or buses.

Signals that remain internal to the worksheet should be labeled as internal, bus members, or comment labels.

To place a module port, select the Module Port command.

Type the module port name at the prompt, then press <ENTER>.DRAFT.
.PN87
.HE PLACE Commands

Returns the following subcommands menu enabling you to specifi the kinds of module ports:Input, Output, Bidirectional, and Unspecified.

Select Input if the module port is used as a signal input.

Select Output if the module port is used as a signal output.

Select Bidirectional if the module port is used as a bidirectional signal.

Select Unspecified if the module port is used to transfer power or "don't care" signals

NOTE

An unspecified module port must be used if power is transferred betwee worksheets. For additonal formation, refer to NETLIST in Section 6.

When you select one of these subcommands, DRAFT returns the module port name to the screen. You may move it whre you want it before placement.

Select Place to place the module port. DRAFT returns you to the "Module Port name?" prompt enabling you to place another module port. Press <ESCAPE> to return to the main command menu.

Figure 4-12 shows the four types of module ports that may be placed in worksheet.

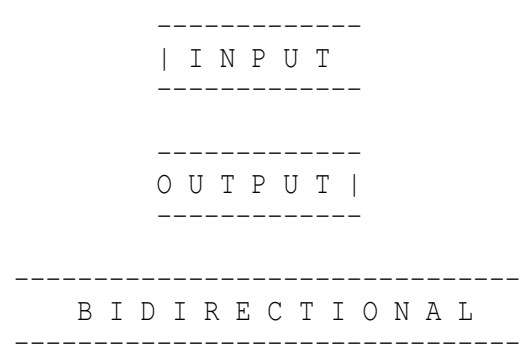


Figure 4-12. Input, Output, Bidirectional and Unspecified Module Ports

NOTE

Module ports renot inteded o b ued as physical connecters, such as DB-9 etc. Physical connectors are objets that should be createdas library parts. For information on working with connctors, see Section 6, the NETLIST utility.

4.12.8 PLACE Power

Select the PLACE power command to place power supply connection on the worksheet.

Select Power. The power object appears on the screen ready to be positioned and placed on the worksheet. DRAFT retuns:

Place Orientation Value Type Find Jump Zoom escape

Select Place to place the power pin where youwant it on the worksheet.

Select Orientation to change the power pin orientation.Select from: Top, Bottom, Left, or right subcommands.

Select Value to change the value used. At the "Power Value?" prompt, erase the previous entry, then type the new entry. For example, +5, +5 VDC, -12 VDC, or any other text string.

Select Type to change the pictorial representation of the power pin. The choice are: circle, arrow, bar, or wave.

Press <ESCAPE> to return to the pin command menu.

NOTE

The power pin default is a circle with a value of VCC. When you execute Place Power pin, the orientatonis forced o thetop. Type and value will be those used previously.

If you use the NETLIST utility program, see Section 6 for information on handling power in battery backup and other applications.

Figure 4-13 shows the four kinds of power objrcts and their orientations.
.pa

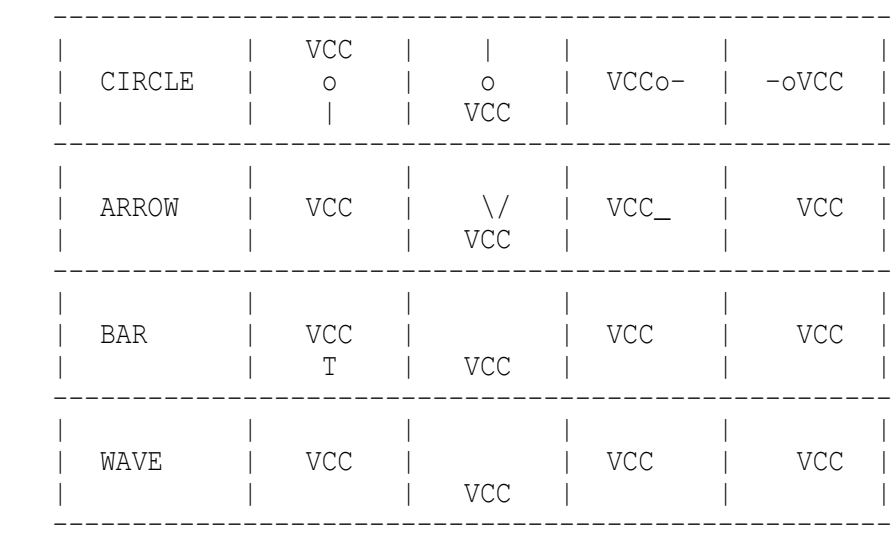


Figure 4.13. Circle, Arrow, Bar, and Wave Power Objects and their Orientations

4.12.9 PLACE Sheet

DRAFT helps you create hierarchical design with sheet symbols. The sheet symbol, reprezenting a workshet in a hierarchy, contains net names used to interconnect the present worksheet and the sheet reprezented by the symbol.

To place a sheet symbol select Sheet. Draft reurns:

Begin Find Jump Zoom escape

Select Begin, outline the area, then select End to finish it.

When you invoke End, DRAFT returns:

Add Delete Edit Name file name Size Zoom escape

Cursor movement is restricted to within the box walls of the hierarchical sheet. This enables you to place the cursor where you want the sheet names and net.

Add

This subcommand is used to add net names to make the connection between worksheets. To add a net name. Select Add Net.DRAFT returns "Net Name?" Type the net name, then press <ENTER>.

Delete

To delete net names, place the cursor at the net name location and select Delete net.

Edit

To edit a net name, place the corsor at the nt name location, then select the Edit net. When the net name appears on the prompt line press <RUBOUT> to erase it, then type the new net name and press <ENTER>.

Name

Use this subcommand to edit the sheet symbol name. The default sheet name is a questions mark(?) located at the top of the sheet. For example: sheet symbol names may be "Memory Array" or "Dynamic RAM Refrish circuitry".

To add a name to the sheet, select Name. DRAFT returns "Sfeet Name?".

Press <RUBOUT> to erase it, then type he desired sheet name.
Press <ENTER> to place it at the top of the sheet.

Filename

The Filename subcommand enables you to name the file representing the hierarchical worksheet. DRAFT automatically generates a filename based on the date and time of day that the sheet was created. This ensures that no two filename will be the same. The DRAFT generated name appears on the prompt line when the file name subcommand is invoked.

If you wish to rename the file, select the file name subcommand and press <RUBOUT> to erase i. TYpe the name that you wish to enter, then press <ENTER> to enter the new file name.

Size

Invoke this subcommand to increase or decrease the sheet size. DRAFT returns:

End Jump Zoom escape

The cursor is automatically positioned on the lower right corner ofbthe sheet. To change sheet size, move the cursor until you reach the desired size, the select end.

Zoom

Invoke this subcommand if you want to zoom to another level (see 4.17, the Zoom command).

escape

Press <ESCAPE> once to return to the edit, find, jump,

zoom, escape subcommand menu, and again to return to the main command menu.

For more information, see 4.5.4, Editing Sheets, and section 5.0 Hierarchy.

4.13 QUIT

Quit enables you to perform several functions depending on the subcommand you select. The Quit command enables you to: enter and leave hierarchical worksheets, load, update, and write to files, clear the worksheet, suspend to DOS, and abandon edits.

When Quit is invoked, DRAFT returns a selection menu:

* Enter Sheet
* Leave Sheet
* Update File
* Write to File
* Initialize
* Suspend to DOS
* Abandon Edit

4.13.1 QUIT Enter Sheet

The Enter Sheet subcommand enables you to enter a hierarchical worksheet, and want to enter another worksheet, select the Enter Shet subcommand.

If you have made any change to the current worksheet, DRAFT returns "Abandon changes made?".

This message tells you that you will lose any change made to the worksheet during teh curent work session. Select "No" to abandon the enter sheet subcommand. If you select "Yes", all changes made to the woksheet are lost, and DRAFT returns:

Enter Leave Find Jump Zoom escape

Place the cursor inside the sheet symbolfor the sheet that you wish to enter and select Enter sheet. For information on saving your latest design session to a file, see 4.13.3, QUIT Update File.

DRAFT remains in the enter sheet mode, enabling you to enter other worksheet. Press <ESCAPE> to return tothe main command menu.

4.13.2 QUIT Leave Sheet

To leave a hierarchical worksheet select the Leave sheet subcommand. When you invoke this command, you move one level up in the command hierarchy. If you areat the top of the command

hierarchy, DRAFT displays an error message briefly on he prompt line telling you thatyou are alredy at the root level.

4.13.3 QUIT Update File

The Update subcommand is used to write your latest worksheet design session to a file. To update a file, select the Update subcommand. If the current worksheet had been previously loaded from a file, that file is updated. If the current worksheet is a unnamed, DRAFT responds with "Write to File?".

Type the desired filename and press <ENTER>.

To update a file other thn the current file, use the QUIT Write File subcommand.

Press <ESCAPE> to return to the main command menu.

4.13.4 QUIT Write File

The Write File subcommand enables you to save th current woksheet to any file you specify. When invoked, DRAFT returns "Write to File?".

Type the desired path and file name, then press<ENTER>. The worksheetis saved to the file specified, and DRAFT returns you the quit subcommand menu.

Press <ESCAPE> to return to the main command menu.

4.13.5 QUIT Initialize

This subcommand enables you to either load a worksheet file or erase everything from it, thus clearingit. To perform these tasks select the QUIT Initialize subcommand and follow one of the two procedures outlined below.

If there are objects on the worksheet, DRAFT returns "Are you sure? Yes or No". Select No to abandon the Initialize subcommand and return to the main command menu. Select Yes to clear the worksheet.

With a clear worksheet, DRAFT return "Load File?". Type the path and filename that you wish to load, then press <ENTER>. If the filename exists, the worksheet is loaded and displayed. If the filename does not exist, DRAFT returns <<<new worksheet>>>.

Press <ESCAPE> to return to the main command menu.

4.13.6 QUIT Suspend to DOSP

This subcommand enables you to temporarily leave DRAFT and the worksheet, save the worksheet in memory, and return to DOS. Once you have suspended DRAFT, you may do other functions that you execute in DOS, including using other software programs as

long as there is enough computer memory.
.pa
To suspend to DOS, select the Suspend to DOS subcommand. DRAFT suspend operation, loads the DOS command interpreter, and adds an additional ">" to the DOS command prompt. This is a reminder that DRAFT is suspended and in the "background". For exemple:

A> This is the DOS prompt that tells you drive A is the default drive.

A>> This is the DOS prompt. DRAFT comes to the "foreground" and the worksheet that you were working on when you suspend DRAFT returns to the screen.

4.13.7. QUIT Abandon Edits

Select this subcommands to exit DRAFT and return to the operating system. If objects have been placed on the worksheet since the last update. DRAFT returns "Are you sure? Yes or No" on the subcommand menu. Select No to abandon the subcommand. Select YesPD to quit and return to the operating system.

.PN95
.HECommands REPEAT

4.14 REPEAT

When the Repeat command is invoked, DRAFT repeats the last entered object or label placed on the worksheet. Repeats are defined in Section 4.15.13, SET Repeat Parameters.

Example:

You want to repeat and auto-increment a bus member label, a unit of one in the Y direction,at 1/10 inch steps. You want to place the labels in the range of A0 through A7. You set the following parameters in the SET repeat parameters command of Section 4.15.13 below.

X Repeat Parameters = + 0

Y Repeat Parameters = + 1

Label Repeat Delta = + 1

Auto Increment Place = Not used in this example

Procedure:

1. Use the PLACE Label command and enter "A0" at the "Label?" prompt.

2. Select bus member as the label type.

3. Place the cursor where the label goes on the worksheet.

4. Place the label.

5. Press <ESCAPE> to return to the main command menu.

6. Press <R> to repeat.

7. Repeat step 6 until you have placed all the labels from A0 to A7.

.PA
.HECommands SET

4.15 SET

With the SET command you may enable or disable these DRAFT options:

* Auto pan

* Making backup files

* Dragging buses when rubberbanding

* Enabling the error bell

* Left mouse button release executes enter

* Macro prompts

* Drawing non-orthogonal wires

* Showing pin numbers

* Disabling the standard title block

* Displaying cursor coordinates, grid dots, and grid references

* Enabling the cursor to move off-grid

* Set the repeat parameters

* Change worksheet size, A through E

When you select the set command, DRAFT returns a subcommand menu displaying the status of each of these options. Worksheet size shows the worksheet size selected. All other options indicate Yes or No.

To change the status of an option, select set. DRAFT returns this selection menu:

* Auto Pan
* Backup File
* Drag Buses
* Error Bell
* Left Button
* Macro Prompts
* Orthogonal
* Show Pins
* Title Block

* Worksheet Size
* X,Y Display
* Grid Parameters
* Repeat Parameters

4.15.1 SET Auto Pan

Auto pan enables movement past the screen boundary. When the cursor crosses a screen boundary, the screen pans in the direction the cursor is moving.

4.15.2 SET Backup File

When the backup file option is enabled, a backup file of your worksheet is created when you write or update files using the QUIT command. The backup file contains the previous version of your edited worksheet.

NOTE

Disabling this option can be dangerous. If your file should accidentally become damaged or erased, you will be unable to recover it.

4.15.3 SET Drag Buses

With this option, buses are rubberbanded when you select a BLOCK drag command. Because there are more points to locate when rubberbanding, system performance decreaseswhen
the BLOCK drag command is executed.

4.15.4 SET Error Bell

With this option, you have the ability to enable or disable the error bell (your computer's speaker). When you enable this option, error messages and errors sound the speaker.

4.15.5 SET Left Button

When you enable this option, releasing the left mouse button executes the <ENTER> command for command line menus only. Pressing the left mouse button continues to execute the command that the video bar highlights.

4.15.6 SET Macro Prompts

With this option enabled, the commands making up your macros are displayed on the screen when the macro is invoked. This option is useful for debugging macros, or to observe the commands being performed when you invoke the macro. For information on macros, see Section 4.11.

4.15.7 SET Orthogonal

When enabled, wires and buses are drawn orthogonally (perpendicular to each other). When disabled, wires and buses may be drawn at any angle.

4.15.8 SET Show Pin Numbers

This option, when enabled, shows library part pin numbers for library parts shown on the screen. When disabled, pin numbers are not shown on the screen. Worksheet hardcopies, in either case, print with pin numbers shown.

4.15.9 SET Title Block

Enabling this option places the standard titleblock on the worksheet. With this option disabled, you may create a custom title block using the ? ACE wire/bus and PLACE label commands.

4.15.10 SET Worksheet Size

This option enables you to select the worksheet size, A through E.

4.15.11 SET X, Y Display

When this option is enabled, the upper right part of the prompt line shows the cursor coordinates. The worksheet origin (0,0) is the upper left corner. Coordinates do not appear on the screen until the cursor is moved.

4.15.12 SET Grid Parameters

The grid parameters command enables you to select grid references, stay on grid, and display visible grid dots. When you select grid parameters, DRAFT returns:

Grid References

When enabled, the grid references option places an alphanumeric border on two of the four worksheet sides shown on the screen. The top border shows grid reference numbers, and the left border reference letters. The borders are scaled to the size of the worksheet.

.pa

Stay on Grid

When enabled, this command confines the cursor to the predefined 1/10 inch grid. When disabled, the cursor may be moved off grid to any position on the worksheet at 1/100-inch resolution.

CAUTION

Placing objects, wires, and buses with this subcommand disabled may cause errors when using ERCHECK and NETLIST. This is because wires and buses may look like they are connected, when in fact, they are not.

ERCHECK and NETLIST may interpret these connections as opens. We recommend that you do not place objects, wires, or buses in the worksheet with this parameter disabled.

Visible Grid Dots

When selected, visible grid dots are displayed on the worksheet spaced 1/10 inch on zoom scale 1,2/10 on zoom scale 2,1/2 inch on zoom scale 5,1 inch on zoom scale 10, and 2 inches on zoom scale 20.

4.15.13 SET Repeat Parameters

This command has four subcommands that are used to set the REPEAT command parameters. The following text explains these subcommands.

X Repeat Step

X repeat step determines the number of unit steps in the X-direction the object being repeated is offset from the original object (the X-direction goes horizontally across the worksheet, with positive to the right and negative to the left of the current cursor position). A unit step is defined as 1/10-inch on grid, and 1/100-inch off-grid.

When you select this subcommand, enter any whole-number integer at the prompt "X Repeat Step?"

Y Repeat Step

Y repeat step determines the number of unit steps in the Y-direction the object being repeated is offset from the original object (the Y-direction goes vertically on the worksheet, with positive down and negative up from the current cursor position). A unit step is defined as ? -inch on-grid, and 1/100-inch off-grid.

When you select this subcommand, enter any whole-number integer at the prompt "Y Repeat Step?"

Label Repeat Delta

This subcommand determines how much the numeric suffix information on a label changes, and in what direction, when the label is repeated. If you enter a positive number, label suffixes are incremented by that number when the repeat command is inwoked. If you enter a negative number, label suffixes are decremented by that number when the repeat command is invoked.

Auto Increment Place

When enabled, this subcommand automatically increments or decrements label names when they are entered on the worksheet using ? place subcommand of PLACE label. After a label has been placed, the numeric suffix of that label is changed by the amount specified by the label repeat delta subcommand.

.pa

4.16 TAG

The Tag command identifies and remembers locations on the worksheet thatare specified by the cursor. You can specify eight locationsach tag is invisible when set (A through H) and use them as destinations for the jump command. Each tag is invisible when set on the worksheet and is not saved with the worksheet.

To set a tag, place the cursor at a location that you want to remember. Then select the Tag command. DRAFT returns:

* A Tag
* B Tag
* C Tag
* D Tag
* E Tag
* F Tag
* G Tag
* H Tag

When the subcommand menu appears on the screen select the tag you want from the menu. Once selected, DRAFT remembers the tag location. Once the tag is set, DRAFT returns you to the main command menu.

EXAMPLE:

1. Place the cursor at a location on the worksheet.
2. Select the tag command from the main command menu.
3. Select tag A from the subcommand menu.
4. Tag location A is now remembered. When Tag A is selected from the jump command, the cursor returns to that location. For more information on the JUMP command, see 4.9, The Jump Command.

.pa

4.17 ZOOM

Zoom enables you to zoom in or out from the worksheet, changing the amount of detail that you see on the screen. You can select five zoom levels, described below.

Scale 1 This most detailed zoom scale is the default.

Scale 2 This second most detailed zoom scale that represents 1/2 of scale 1.

Scale 5 The third most detailed zoom scale that represents 1/5 of scale 1.

Scale 10 The fourth most detailed zoom scale that represents 1/10 of scale 1.

Scale 20 The least detailed zoom scale that represents 1/20 of scale 1.

To change the zoom scale select zoom. DRAFT returns:

* Center
* In
* Out
* Select

4.17.1 ZOOM Center

When selected, the ZOOM center subcommand re-centers the displayed portion of the sheet around the cursor. This subcommand is useful for centering an object on the screen so you can easily edit it.

For example: If an object is displayed partially off the screen, you may center it by placing the cursor near the object and selecting ZOOM Center.

The number in parentheses shows the current zoom scale (1-
(1-20).

4.17.2 ZOOM In

The ZOOM In subcommand enables you to zoomin on on the worksheet for a more detailed view. The number in parentheses shows you what the zoom scale will be the next time you invoke ZOOM in

.pa

4.17.3 ZOOM Out

The ZOOM out subcommand enables you to zoom out to display a ? worksheet area. The number in parentheses shows you what the zoom scale will be the next time you invoke ZOOM out

4.17.4 ZOOM Select

The Select subcommand enables you to select any one of five zoom scales from a pop-up subcommand menu. When invoked, you can select zoom scale 1,2,5,10, or 20 from the menu.
.pn104

5.H I E R A R C H Y

In section 3 you were introduced to the three types of file structures used to create schematics with OrCAD/SDT. These are: flat file, hierarchy, and once sheet file structures.

The flat file structure is the traditional method for creating a complex design that contains many worksheet. Each file design must be created as a collection of one sheet schematics. This is not only time consuming, but it also becomes a unmanageable task as the number of worksheets increases in your design. In a flat file design, you have to keep of track the worksheet file names.

In a hierarchy, each individual file name is maintaind by DRAFT. Each worksheet is organized as a block, or sheet symbol, which contains part of the design. These sheet symbols can contain progressively more detail and may be used in multiple instances to replicate common circuit functions. Hierarchy not only makes the design pocess easier, but it also provides the framework necessary to keep the design more manageable. Many of the benefits of a hierarchical design over a flat file are listed below and will be discussed in detail throughout this section.

* Files are easier to manage
* Worksheets may be used repetitively in the design
* Utility programs are easier to implment
* Designs are created faster and easier.

The introduction to hierarchical file structures begins by introducing you to the elements of hierachical design. This follows with a discussion of the two types of hierarchical designs: simple and complex. Finally, you will learn how to create a simple and complex hierarchy.

5.1. ELEMENTS OF A Hierarchy

In a hierarchical structure, these is one "root" worksheet. This is the first level in the hierarchy. By using the PLACE Sheet command, individual blocks are placed in the root worksheet. These blocks are known as "sheet symbols" and they represent separate unique worksheet. To display each sheet symbol as o worksheet, just place the cursor inside the sheet symbol boundary and execute the QUIT Enter command. This places you one level down in the hierarchy.

When you enter a sheet symbol, the worksheet that represents that sheet symbol appears on the screen. If other sheet symbols are placed in the worksheet and you continue to enter them, anoher worksheet will appear on the screen. This hierarchical procedure can continue to a depth of over 5000 levels with OrCAD/SDT.

Connections between scheet symbols are made through nets that are placed along the left and right sides of the sheet symbol. As in a flat file structure, module ports are used in a hierarchy to label signals that leave a worksheet. In a hierarchy, nets are used to connect module ports to a sheet symbol.

Listed below are many of the terms and definitions that are used in creating a hierarchical design. As in a flat file structure, it is important to follow these procedures.

N O T E

We recommend that you refer to NETLIST in Section 6, for more information on many of the subjects discussed here.

ROOT SHEET

The "root" sheet is the first worksheet in the hierarchy. When loading a hierarchical file into DRAFT or executing a utility program, you specify the root file name.

SHEET SYMBOL

Sheet symbols are drawn in a worksheet to represent a separate worksheet. Figure 5-1 shows an example of three sheet symbols. To place a sheet symbol in a worksheet, use the PLACE Sheet command.

A sheet symbol is tipically assigned a "Name", which identifies its function. For example, the sheet symbols in Figure 5-1 are named CPU, I/O and MEMORY. Since a sheet symbol represents a worksheet, file name should also be assigned. A "Filename" is automatically assigned to the sheet symbol based on the date and time of day. The filename may be modified if you do not want to use the one assigned by DRAFT. For additional information on placing sheet symbols, see Section 5.2. on the PLACE Sheet command.

(No Figure !)

Fig. 5-1. A Root Worksheet with Three Sheet Symbols

.pa

INTERNAL LABELS

As in the flat structure, internal labels are important in a hierarchy. An internal label must be placed on every bus that is used in a worksheet. This informs the NETLIST utility porgram how many members are associated with a bus.

Internal bus labels must be in the form:

BUSNAME[0..n]

Where BUSNAME is called the "prefix" and represents the name of the bus. [0..n] is called the "suffix", where n represents the decimal number of the last member of the bus; only a zero (0) is valid in the first portion of the suffix ([2..n] for example is not valid). Between the prefix and the suffix there must be no

space.
Typycal examples are:

```
ADDR[0..31]    (this bus has 32 members)
DATA[0..15]    (this bus has 16 members)
CONTROL[0..3]  (this bus has  4 members)
```

BUS MEMBER LABELS

Bus member labels are used to label the individual members that come from a bus. As a rule, you may place any number of bus member labels on a bus member. Since bus member labels are associated with a bus, they must be labeled in a form that corresponds to the bus they come from. This form is:

BUSNAMEx

Where BUSNAME is the same prefix name as the internal bus label, x a decimal number in the range of [0-n] taken from the suffix of the internal bus label, and n represents the decimal member of the last member of the bus. Between BUSNAME and x there must be no space.

MODULE PORTS

Any signal that leaves the worksheet must does so via a module port. In a hierarchical structure, the signals that connect to a particular worksheet do so through module ports. A module port may have any name when it is connected to a wire. However, when connected to a bus, a module port must be in the form:

BUSNAME[0..n]

Where BUSNAME is called the "prefix" and it represents the name of the bus. [0..n] is called the "suffix", where n represents the decimal number of the last member of the bus. Between the prefix and the suffix there must be no space.

A typical example would be:

```
ADDR[0..3]
DATA[0..15]
CONTROL[0..3]
```

NETS

Nets are connections that are made from a sheet symbol to other objects or sheet symbols. In figure 5-1, above, the sheet symbol named I/O has two nets: A[0..7] and C[0..3]. In this example, these nets connect to buses, which in turn connect to other sheet symbols.

In a hierarchy, nets make"implied" connections to their respective module ports. For example, in Figure 5-1, the sheet symbol named I/O has two nets: A[0..7] and C [0..3]. If you enter the sheet symbol (using the QUIT Enter command) to obtain the I/O worksheet, there must also be two module ports named: A[0..7] and

C[0..3].

To summarize, net must have the same name as the module port that connects to it in the sheet symbol.

QUIT ENTER SHEET

The QUIT Enter Sheet command enables you to enter a hierarchical sheet symbol to obtin the corresponding worksheet. To enter sheet symbol, select the QUIT Enter command from the main menu. Place the cursor inside the sheet symbol that you want to enter. Then, select the Enter subcommand.

QUIT LEAVE SHEET

The QUIT Leave Sheet command enables you to leave worksheet and return one level up in the hierachy to the sheet symbol. To leave worksheet, select the QUIT Leave Sheet command from the main command level, then select the Leave subcommand.

5.2. THE PLACE SHEET COMMAND

DRAFT enables you create hierarchical designs with sheet symbols. The sheet symbol, representing a worksheet in a hierarchy, contains net names used to interconnect the present worksheet and the sheet represented by the symbol.

To place a sheet symbol select the PLACE sheet command. DRAFT returns:

Begin Find Jump Zoom escape

Select Begin, outline the area, then select End to finish it. When you invoke End, DRAFT returns:

Add Delete Edit Name file name Size Zoom escape

Cursor movement is restricted to within the box walls of the hierarchical sheet. This enbles you to place the cursor where you want the sheet names and nets.

Add

This subcommand is used to add net names to make the connections between worksheets. To add a net name, place the cursor at the edge of the box where you want the name.

Select Add net. DRAFT returns "net Name?" Type the net name, then press <ENTER>.

Delete

To delete net names, place the cursor at the net name location and select Delete net.

Edit

To edit a net name, place the cursor at the net name location, then select Edit net.

When the net name appears on the prompt line press <RUBOUT> to erase it, then type the new net name and press <ENTER>.

Name

Use this subcommand to edit the sheet symbol name. The default sheet name is a question mark (?) located at the top of the sheet. For example: sheet symbol names may be "Memory Array" or "Dynmic RAM Refresh circuitry".

To add a name to the sheet, select Name. DRAFT returns "Sheet Name?"

Press <RUBOUT> to erase it, then type the desired sheet name. Press <ENTER> to place it at the top of the sheet.

To edit a sheet name select Name and erase the previous name from the prompt line. Type the new sheet name and press <ENTER> to place it at the top of the sheet.

Filename

The Filename subcommand enables you to name the file representing the hierarchical worksheet. DRAFT automatically generates a filename based on the date and time of day that the sheet was created. This ensures that no two filenames will be the same. The DRAFT-generated name appears on the prompt line when the Filename subcommnd is invoked.

If you wish to rename the file, select the Filename subcommand and press <RUBOUT> to erase it. Type the name that you wish to enter, then press <ENTER> to enter the new filename.

Size

Invoke this subcommand to increase or decrease the sheet size. DRAFT returns:

Enter Jump Zoom escape

The cursor is automatically postioned on the lower right corner of the sheet. To change sheet size, move the cursor until you reach the desired size, then select End.

5.3. TYPES OF HIERARCHICAL DESIGNS

When creating a hierarchy, there are two types designs that you can make: a simple hierarchy and a complex hierarchy. Both types are created in the same manner, which is illustrated in examples later in this Section.

A simple hierarchy is one in which all sheet symbols are unique. Any number of sheet symbols can be used in the design. Connectivity between sheets is made through nets and module ports.

A complex hierarchy is one in which some sheet symbols are used multiple times in the design. This aspect of hierarchical organization is the ability to replicate blocks of common logic. This means that circuitry may be placed into worksheet, a sheet symbol is drawn to represent that worksheet, and that sheet symbol is placed in multiple instances within other worksheets. Like the simple hierachy, a complex uses nets and module ports to connect signals from one worksheet to another.

5.4. EXAMPLES OF A SIMPLE HIERARCHY

In this section, we will describe an example of a simple hierarchical design, discuss labeling, module ports, nets, sheet

symbols, and other aspects of the design, and review examples on executing some of the utility programs.

The example schematic discussed is a three sheet simple hierarchy. It is defined as a simple hierarchy because none of the sheet symbols are replicated and used in multiple instances. The root sheet is illustrated in Figure 5-2 below, which has the title: CMOS CPU DESIGN.

(No Figure !)

Figure 5-2. CMOS CPU DESIGN Root Sheet

When you create a hierarchy, the first step is to create the first sheet, or "root". This sheet is create like any other schematic. For this example, the root sheet is given the first name: CMPSCPU.SCH. As Figure 5-2 shows, the name of the worksheet is: CMOS CPU DESIGN, and appears in the title block.

Inside of the root sheet is the design's circuitry. In this case, the 80C51, 82C82, and discrete components. Also placed in th root sheet, are two sheet symbols: POWER SUPPLY and CMOS MEMORY.

These sheet symbols were placed in the worksheet using the PLACE Sheet command. The CMOS MEMORY sheet symbol contains the worksheet in which the system memory is located. The POWER SUPPLY sheet symbol contains the worksheet in which the systems power supply is located.

When you create a sheet symbol, DRAFT automatically assigns it a file name based on the date and time of day. In this example, we have modified the default file names and assignd the CMOS MEMORY sheet symbol the file name: MEMORY.SCH. The POWER SUPPLY sheet symbol is assigned the file name: POWER.SCH.

The CMOS MEMORY sheet symbol contains four nets: A[0..7] WE, BACKUP, and AD[0..7]. Connected to the A[0..7] and AD[0..7] nets, are buses that have internal labels placed on them with the same name as the net. As mentioned earlier and in the NETLIST utility description in Section 6, this labeling procedure is mandatory for buses. Connected to the WE net is a wire that goes to the PSEN signal on the 80C51. Last, a net named BACKUP is connected to a net in the POWER SUPPLY sheet symbol having the same name.

For labels and module ports, there should be no space between the prefix and suffix.

Finally, in the root sheet is a power object named +5V connected to a power object named VD. This connects the VDD pins of the 80C51 and the 82C82 to the plus 5 volt supply. Likewise, a power object named GND is connected to power object named VSS. This connects the VSS pin of the 80C51 and the 82C82 to power ground symbols.

After the root design is completed, the QUIT Update command should be used. In this example, you would be updating the root file name: CMOSCPU.SCH.

The next step is to enter one of the hierarchical sheet symbols in the root sheet to complete the design. To do this, select the QUIT Enter Sheet command. Place the cursor inside the POWER SUPPLY sheet symbol and select the Enter subcommand. Once this is done, you will be inside new worksheet to be used as the power supply. The completed design is shown in fgure 5-3 below.

(No Figure !)

Figure 5-3. POWER SUPPLY Worksheet

Inside the POWER SUPPLY sheet symbol is the power supply circuitry. Notice the module port named BACKUP. This connects to the net named BACKUP in the POWER SUPPLY sheet symbol of Figure 5-3 above. Electrically, the BACKUP module port connects to the module port, of the same name in the CMOS MEMORY sheet symbol.

In operation, the CMOS MEMORY sheet only receives DC power via the module port named BACKUP. Power is isolated in the CMOS MEMORY worksheet, because the power is transferred through the module port, (refer to the NETLIST discussion in Section 6).

When the power supply design has been completed, the file should be updated before you leave the worksheet. Update the file by executing the QUIT Update command. Once updated, you may leave the hierarchical sheet and return to the root by executing the QUIT Leave Sheet command.

Back at the root level, the next step is to enter the CMOS MEMORY Enter Sheet symbol and complete the design. To do this select the QUIT Enter Sheet command, place the cursor inside the sheet symbol and execute the Enter subcommand. Once this is done, you will be inside a new worksheet to be used as the CPU's memory. The completed design is illustrated in figure 5-4 below.

(No Figure !)

Figure 5-4. CMOS MEMORY Worksheet

Inside the CMOS MEMORY sheet symbol is the memory section associated circuitry. In this worksheet are four module ports: A[0..7], WE, BACKUP, and AD[0..7]. They connect to identically named nets located in the CMOS MEMORY sheet symbol of figure 5-4 below.

As required, the buses that connect to module ports are named with internal labels having the same name as the module port. In this case, module port A[0..7] connects to a bus labeled A[0..7]. Module port AD[0..7] connects to a bus labeled D[0..7] (notice that the suffix has the same name, but the prefix differs).

Also required in a hierarchical design, bus member names must have the same prefix name as the bus that they connect to, for example, bus members D0 to D7 correspond to the prefix of the bus labeled D[0..7] (the prefix is a "D" in this case). Likewise, bus members A0 through A7 correspond to the prefix of the bus labeled A[0..7] (the prefix is an "A" in this case).

DC power in supplied to the CMOS MEMORY worksheet through the module port named BACKUP. As mentioned, power is isolated in this worksheet since the VDD power object is connected to the modul port named BACKUP.

Finally, a power object named GND is connected to a power object named VSS. This connects the VSS pins of the 51C68 memory devices to the power ground symbol that the C\S\ pins are connected to.

When design work is completed on the CMOS MEMORY worksheet, it should be updated before leaving the hierarchy.

This complets the discussion of a simple hierarchy. In the following Sections, the design will be run through the ANNOTATE, ERCHECK, TREELIST, NETLIST and PARTLIST utility programs.

5.4.1.Invoking The ANNOTATE Utility On CMOSCPU.SCH

After the design has been completed, it should be run through ANNOTATE Utility program (this assumes that you have not manually annotated the worksheet). To annotate the simple CMOSCPU.SCH hierarchy, enter the following from the DOS command line:

ANNOTAE CMOSCPU. SCH /M <ENTER>

ANNOTATE is the name of the utility, CMOSCPU.SCH is the name of the root worksheet of the hierarchy, the /M switch merges the annotation information into each hierarchical worksheet, <ENTER> refers to pressing the ENTER key.

After the utility program has completed, the worksheet is annotated. Next, errors in the design are checked for using the ERCHECK program.

5.4.2. Invoking the ERCHECK Utility On CMOSCPU.SCH

Generally, every design that you create should be checked for errors using the ERCHECK utility program after the worksheet has been annotated. To execute the ERCHECK utility on the simple CMOSCPU.SCH hierarchy, enter the following from the DOS command line:

ERCHECK CMOSCPU.SCH ERROR.TXT <ENTER>

ERCHECK is the name of the utility, CMOSCPU.SCH is th name of the root worksheet of the hierarchy, ERROR.TXT is the name of a text file to send the error information (this is useful for examining the results of the utility), <ENTER> refers to pressing the ENTER key.

After the utility program has completed, all errors are sent to the ERROR.TXT file. To examine the file, use a text editor or word processing software. Examining the ERROR.TXT file results in the following:

```
WARNING - POWER Supplies are CONNECTED VDD <-> + 5 V
WARNING - POWER Supplies are CONNECTED VDD <-> GND
WARNING - INPUT has NO Driving Source U4,AB
WARNING - INPUT has NO Driving Source U4,A9
WARNING - INPUT has NO Driving Source U4,A10
WARNING - INPUT has NO Driving Source U4,A11
WARNING - INPUT has NO Driving Source A0
WARNING - INPUT has NO Driving Source A1
WARNING - INPUT has NO Driving Source A2
WARNING - INPUT has NO Driving Source A3
WARNING - INPUT has NO Driving Source A4
WARNING - INPUT has NO Driving Source A5
WARNING - INPUT has NO Driving Source A6
WARNING - INPUT has NO Driving Source A7
WARNING - INPUT has NO Driving Source U5,AB
WARNING - INPUT has NO Driving Source U5,A9
WARNING - INPUT has NO Driving Source U4,A10
WARNING - INPUT has NO Driving Source U4,A11
WARNING - POWER Supplies are CONNECTED VSS <-> GND
```

Discussion:

The ERCHECK program checks for a number of problems associated with a design. This includes: open input pins, shorts, and bus contention.

A WARNING is issued to inform you of certain conditions that may be overlooked when creating your design. They are not critical errors. In this example, most of the warnings inform you of inputs with no driving source. This is perfectly acceptable, since we intentionally left these pins open in the design. The connected power supply warnings are also acceptable, since they were intended to be connected in the design.

If you obtain ERRORS in a ERCHECK output, they should be corrected before continuing on and running other utility programs.

In this example, all warnings are acceptable and other utilities may be run.

.PN114

5.4.3. Invoking The TREELIST Utility On CMOSCPU.SCH

To obtain a text file tree list of a hierarchy, use the TREELIST utility. This program is helpful in organizing a hierarchy that contains many worksheets. To execute the TREELIST utility on the simple CMOSCPU.SCH hierarchy, enter the following from the DOS command line:

```
TREELIST CMOSCPU.SCH TREE.TXT <ENTER>
```

TREELIST is the name of the utility, CMOSCPU.SCH is the name of the root worksheet of the hierarchy, TREE.TXT is the name of a text file to send the tree information (this is useful for examining the results of the utility),<ENTER> refers to pressing the ENTER key.

After the utility has completed,the tree information is sent to the TREE.TXT file. To examine the file, use a text editor or word processing software. Examining the TREE.TXT file results in the following:

```
<<<Root File>>>
[CMOSCPU.SCH]
   POWER SUPPLY
   [POWER.SCH]
   CMOS MEMORY
   [MEMORY.SCH]
```

Discussion:

All worksheet file names are enclosed within brackets [file names], sheet symbol names are listed above the file names. In this example, we see the root file has a file name of: CMOSCPU.SCH. Below the root, are the remaining sheet symbols and their associated file names. File POWER.SCH belongs to the sheet symbol named POWER SUPPLY. File MEMORY.SCH belongs to the sheet symbol named CMOS MEMORY.

5.4.4.Invoking The NETLIST Utility On CMOSCPU.SCH

To obtain a netlist of a hierarchy, use the NETLIST utility. For this example, a CALAY net list will be extracted. To extract

a CALAY NETLIST of CMOSCPU.SCH hierarchy, enter the following from the DOS command line:

```
NETLIST CMOSCPU.SCH NET.TXT CALAY /S <ENTER>
```

NETLIST is the name of the utility, CMOSCPU.SCH is the name of the root worksheet of the hierarchy, NET.TXT is the name of a text file to send the net information (this is useful for examining the results of the utility), CALAY is the name of the desired net list format, /S indicates a special net list format is desired (CALAY in this case), <ENTER> refers to pressing the ENTER key.

After the utility program is completed, the net information is sent to the NET.TXT file. Since a CALAY net list format was specified, a component file is also generated. The component file generated has a CMP extension added to the NET text file. To examine the file, use a text editor or word processing software. Exmining the NET.TXT file and NET.CMP file, results in the following:

NET.TXT

```
/N00001 U1(40) U2(20);
/N00002 C2(1) X1(1) U1(19);
/N00003 X1(2) C1(1) U1(18);
/N00004 U1(4) Q1(COLLECTOR) R2(2);
/N00005 U2(11) U1(30);
/N00006 R1(2) SW1(NO) C3(1) Q1(BASE);
/N00007 T1(BA) D4(AC IN 1);
/N00008 Q2(COLLECTOR) D4(DC OUT PLUS) R3(1) C4(1);
/N00009 Q2(BASE) D1(CATHODE) U3(VI);
/N00010 R3(2) D1(ANODE);
/N00011 T1(BB) D4(AC IN 2);
/N00012 U3(GND) R4(2);
/N00013 BT1(+) D2(ANODE);
/N00014 U4(20) U5(20);
/N00015 U5(8);
/N00016 U5(7);
/N00017 U5(6);
/N00018 U5(5);
/N00019 U5(4);
/N00020 U5(3);
/N00021 U5(2);
/N00022 U5(1);
/N00023 U1(29) U4(11) U5(11);
/VDD    R2(1) R1(1) U3(VO) Q2(EMITTER) D3(ANODE) C5(1);
/VSS    SW1(COMMON) Q1(EMITTER) C3(2) U2(9) C1(2) C2(2)
        U1(31) U2(10),
        U1(20) BT1(-) D4(DC OUT MINUS) C4(2) C5(2) R4(1)
        R4(WIPER) U4(9) U5(9) U4(10) U5(10);
/N00026 U2(19) U4(8);
/N00027 U2(18) U4(7);
/N00028 U2(17) U4(6);
/N00029 U2(16) U4(5);
/N00030 U2(15) U4(4);
/N00031 U2(14) U4(3);
/N00032 U2(13) U4(2);
/N00033 U2(12) U4(1);
/N00034 U2(1) U1(39) U4(15);
/N00035 U2(2) U1(38) U4(14);
```

```
/N00036 U2(3) U1(37) U4(13);
/N00037 U2(4) U1(36) U4(12);
/N00038 U2(5) U1(35) U4(15);
/N00039 U2(6) U1(34) U4(14);
/N00040 U2(7) U1(33) U4(13);
/N00041 U2(8) U1(32) U4(12);
/N00042 D3(CATHODE) D2(CATHODE);
```

NET.CMP CALAY Component text file

```
10          R3          shape     - X -     - Y -     0
10 UF       C3          shape     - X -     - Y -     0
10 K        R2          shape     - X -     - Y -     0
12 MHZ      X1          shape     - X -     - Y -     0
1K          R4          shape     - X -     - Y -     0
1N4001      D2          shape     - X -     - Y -     0
1N4001      D3          shape     - X -     - Y -     0
21K         R1          shape     - X -     - Y -     0
30 PF       C1          shape     - X -     - Y -     0
30 PF       C2          shape     - X -     - Y -     0
4 V         BT1         shape     - X -     - Y -     0
47          C4          shape     - X -     - Y -     0
47          C5          shape     - X -     - Y -     0
4           T1          shape     - X -     - Y -     0
51C68       U4          shape     - X -     - Y -     0
51C68       U5          shape     - X -     - Y -     0
80C51       U1          shape     - X -     - Y -     0
82C82       U2          shape     - X -     - Y -     0
CR0127      D4          shape     - X -     - Y -     0
IN4004      D1          shape     - X -     - Y -     0
LM123       U3          shape     - X -     - Y -     0
NPN         Q1          shape     - X -     - Y -     0
SPST SW1                shape     - X -     - Y -     0
TIP31C      Q2          shape     - X -     - Y -     0
```

5.4.5. Invoking The PARTLIST Utility On CMOSCPU.SCH

To obtain a text file part list of a hierarchy, use the PARTLIST utility. To execute the PARTLIST utility on the simple CMOSCPU.SCH hierarchy, enter the following from the DOS command line:

```
PARTLIST CMOSCPU.SCH PART.TXT <ENTER>
```

PARTLIST is the name of the utility, CMOSCPU.SCH is the name of the root worksheet of the hierarchy, PART.TXT is the name of a text file to send the part information (this is useful for examining the results of the utility), <ENTER> refers to pressing the ENTER key.

After the utility program has completed, the part information is sent to the PART.TXT file. To examine the file, use a text editor or word processing software. Examining the PART.TXT file, results in the following:

```
.pa
CMOD CPU DESIGN                        Revised: October 26, 1986
123456789                              Revision:A
Bill of Materials October 01, 186      15:51:49          Page 1

Item   Quantity     Reference                        Part
```

1	1	X1	12 mHz
2	2	C1,C2	30 pf
3	1	Q1	NPN
4	1	C3	10 UF
5	1	R1	2.7 K
6	1	R2	10 K
7	1	U1	80C51
8	1	U2	82C82
9	1	SW1	SPST
10	1	D1	IN4004
11	1	R3	10
12	2	C4,C5	470
13	1	Q2	TIP31C
14	1	R4	1K
15	1	BT1	4 V
16	2	D2,D3	1N4001
17	1	U3	LM123
18	1	T1	4:1
19	1	D4	CR0127
20	2	U4,U5	51C68

5.5. COMPLEX HIERARCHIES

As mentioned previously, a complex hierarchy is one in which some sheet symbols are used multiple times in the design. This type of hierarchical organization enables you to replicate blocks of common logic without having to create a separate worksheet. All you do is place the circuit to be replicated into a worksheet. Then draw a sheet symbol to represent that worksheet, and place that sheet symbol in multiple instances within other worksheets.

Creating a complex hierarchy is just easy as creating a simple hierarchy. The difference between the two types comes when handling and executing utility programs. A simple hierarchy stores the reference designators in each worksheet. However, a complex hierarchy uses one or more worksheets multiple times. Therefore, you cannot store the reference designators in the worksheets. In this case, an annotation file provides the method for storing the reference information in a binary format.

The annotation file is created using the ANNOTATE utility program. When you use other utilities, the annotate file is used as the input file instead of the root worksheet file name.

.pa

5.6. AN EXAMPLE OF A COMPLEX HIERARCHY

In this section, we will describe an example of a complex hierarchical design. The example begins with a design that has not been run through the ANNOTATE utility. Labeling. module ports, nets, sheet symbols, and other aspects of the design are discussed. Finally, we will review creating an annotation file and executing many of the utility programs.

The example schematic discussed is a thirteen sheet complex hierarchy. It is defined as a complex hierarchy because many of the sheet symbols are replicated and used in muliple instances throughout the design. As will be demonstrated later, you are only required to create three separate worksheets to produce the

thirteen sheet hierarchy. The example root worksheet is illustrated in Figure 5-5 below and called 4BIT ADDER.

(No Figure !)

Figure 5-5. 4BIT ADDER Root Sheet

As you examine the root worksheet, notice that there are four identical sheet symbols placed. They are named fulladd. Module ports are used to connect signals from the outside world the appropriate sheet symbol and net.

Since all four full adders are identical in their design, you do not have to create a separate worksheet for each of them in a complex hierarchy; you create one.

By entering any one of the four full adder sheet symbols (QUIT Enter Sheet command) you will be inside of a new worksheet. This worksheet contains the design of the full adder which is shown in Figure 5-6 below.

(No Figure !)

Figure 5-6. FULL ADDER Worksheet

As you examine the contents of this worksheet, notice that there are two new sheet symbols named halfadd. Each one is identical. Each module port in the FULL ADDER worksheet is appropriately named to connect to the nets in the 4BIT ADDER worksheet, one level up in the hierarchy.

Since the two half adder sheet symbols are identical, you do not have to create a separate worksheet for each of them in a complex hierarchy.

Finally, by entering one of the half adder sheet symbols, you will be inside of a new worksheet. This worksheet contains the design of the half adder which is shown in in Figure 5-7. below.

(No Figure !)

Figure 5-7. HALF ADDER Worksheet

As you examine the contents of this worksheet, notice that each module port in the HALF ADDER worksheet is appropriately named to connect to the nets in the FULL ADDER worksheet, one level up in the hierachy.

5.6.1. Power Objects

This section discusses the placement of the ground and power objects in each of the three worksheet. As Section 6 discusses in the NETLIST utility, power pins in a library part are by default a local signal. They create a net where all of the similar power pin names are connected together when you generate a net list (all VCC's are connected, GND's are connected, VDD's are connected etc). Each type of power pin is isolated in the worksheet where is placed. To get them off of a worksheet to connect to another worksheet, you have to choose how it is going to be done. Power does not leave the worksheet automatically.

To make power global to transfer it off of a worksheet and

connect it to power in another worksheet, a power object must be placed in each worksheet. One for VCC, GND, VDD, VSS etc. This is true in any type of worksheet file structure. Power objects are placed in a worksheet using the PLACE Power command, or by placing grounds from the DEVICE library.

This is the purpose of the VCC and GND power objects that are placed in each of the three hierarchical worksheets. They connect power from eah worksheet together. It does not matter if a power object is physically connected to another object. At least one power object of each type has to be placed in a worksheet for the ne list program to connect all power signals together.

5.6.2. Further Discussion Of The Complex Hierarchy Example

As you may have noticed, only three separate worksheets have been created. However, after examining the root worksheet, there are fourseparate full adders. Each full adder sheet symbol consists one full adder worksheet and two half adder worksheets. This results in three total worksheets for each full adder sheet symbol. The total number of worksheets for the complex 4Bit hierarchy is 13 worksheets x 4 full adders - 1 for the root worksheet = 13 worksheets.

Since complex hierarchy uses one or more worksheets multiple times, you cannot store the reference designators in each of the repetitively used worksheets. In the 4Bit Adder example, only three worksheets were created. For complex hierarchies, an annotation file provides the method for storing the reference information for each of the additional worksheets that have not been created.

The annotation file is cerated using the ANNOTATE utility program. When other utilities are used, thr annotation file is used as the input file instead of the root worksheet file name that was used in a simple hierarchy.

.pa

5.6.3.Creating the Annotation File

After the complex hierarchy design has been completed, it should be run througn ANNOTATE utility program to generate the annotation file. To annotate the complex hierarchy, assume that the root worksheet file name is: 4BIT.SCH. Enter the following from the DOS command line:

```
ANNOTATE 4BIT.SCH 4BIT.ANN <ENTER>
```

ANNOTATE is the name of the utility, 4BIT.SCH is the name of the root worksheet of the hierarchy, 4BIT.ANN is the name of the annotation file, <ENTER> refers to pressing the ENTER key.

After the utility program has completed, an annotation file is generated which will be used as the source in all other utilities. Next, errors in the design are checked for using the ERCKECK program.

5.6.4. Invoking The ERCHECK Utility

To run the ERCHECK utility program, specify the annotation file: 4BIT.ANN as the source. Enter the following from the DOS commands line:

```
ERCHECK 4BIT.ANN ERROR.TXT /A <ENTER>
```

ERCHECK is the name of the utility, 4BIT.ANN is the name of the Annotation file, ERROR.TXT is the name of a text file to send the error information (this is useful for examining the results of the utility), /A signifies that the source is an annotation file, <ENTER> refers to pressing the ENTER key.

In this example, the ERROR.TXT file will contain no errors. However, WARNINGS will be generated to the screen informing you that there is "Unconnectd POWER and GND". This is because the power objects tha were placed in each of the three worksheets are not connected to any other object. For this example, the warning can be ignored. You must place a power object in the workscheets to transfer power from sheet to sheet.

5.6.5. Invoking The TREELIST Utility

To obtain a text file tree list, use the TREELIST utility. This program is helpful in organizing a hierarchy that contains many worksheets. To execute the TREELIST utility, specify the root hierarchy file: 4BIT.SCH as the source-not the annotation file. Enter the following from the DOS command line:

```
TREELIST 4BIT.SCH TREE.TXT <ENTER>
```

TREELIST is the name of the utility, 4BIT.SCH is the name of the root worksheet of the hierarchy, TREE.TXT is the name of a text file to send the tree information (this is useful for examining the results of the utility), <ENTER> refers to pressing the ENTER key.

After the utility program has completed, the tree information is sent to the TREE.TXT file. To examine the file, use a text editor or word processing software. Examining the TREE.TXT file results in the following:

```
<<<Root File>>>
[4BIT.SCH]
  fuladd
  [fuladd.sch]
    halfadd
    [halfadd.sch]
    halfadd
    [halfadd.sch]
  fuladd
  [fuladd.sch]
    halfadd
    [halfadd.sch]
    halfadd
    [halfadd.sch]
  fuladd
  [fuladd.sch]
    halfadd
    [halfadd.sch]
    halfadd
    [halfadd.sch]
  fuladd
  [fuladd.sch]
    halfadd
    [halfadd.sch]
    halfadd
```

[halfadd.sch]

Discussion:

All worksheet file names are enclosed within brackets [file names], sheet symbol names are listed above the file names. In this example, we see the root file name of: 4BIT.SCH. Below the root, are the remaining sheet symbols and their associated file names.

5.6.6. Invoking The NETLIST Utility

To obtain a net list of a hierarchy, use the NETLIST utility. For this example, a COMPUTERVISION net list will be extracted. To extract a COMPUTERVISION NETLIST, specify the annotation file: 4BIT.ANN as the source. Enter the following from DOS command line:

NETLIST 4BIT.NN NET.TX COMPUTERVISION/A/S <ENTER>

NETLIST is the name of the utility, 4BIT.ANN is the name of the Annotation file, NET.TXT is the name of a text file to send the net information (this is useful for examining the result of the utility), COMPUTERVISION is the name of the desired net list format, /A specifies that the source is Annotation file, /S indicates a special net list format is desired, <ENTER> refers to pressing the ENTER key.

After the utility program has completed, the net information is sent to the NET.TXT file. To examine th file, use a text editor or word processing software. Examining the NET.TXT file results in the following:

NET.TXT

```
0001 N00001              U2-6      U1-4
0002 N00002              U2-5      U3-3
0003 N00003              U1-5      U2-3
0004 N00004              U3-4      U2-2
0005 N00005              U4-3      U1-9
0006 N00006              U4-2      U3-6
0007 N00007              U1-10     U2-11
0008 N00008              U3-8      U2-13
0009 N00009              U4-11     U6-1
0010 N00010              U4-13     U3-10
0011 N00011              U6-2      U4-8
0012 N00012              U3-12     U4-10
0013 N00013              U5-8      U6-4
0014 N00014              U5-10     U7-2
0015 N00015              U6-5      U5-6
0016 N00016              U7-4      U5-5
0017 N00017              U8-6      U6-12
0018 N00018              U8-5      U7-6
0019 N00019              U6-13     U8-3
0020 N00020              U7-8      U8-2
0021 N00021              U9-3      U10-1
0022 N00022              U9-2      U7-10
0023 N00023              U10-2     U8-11
0024 N00024              U7-12     U8-13
0025 N00025              U9-11     U10-9
```

```
     0026 N00026             U9-13     U12-2
     0027 N00027             U10-10    U9-8
     0028 N00028             U12-4     U9-10
     0029 N00029             U11-8     U10-12
     0030 N00030             U11-10    U12-6
     0031 N00031             U10-13    U11-6
     0032 N00032             U12-8     U11-5
     0033 N00033             U10-5     U11-11
     0034 N00034             U6-10     U9-6
     0035 N00035             U1-13     U5-11
     0036 N00036             U1-2      U4-6
     0037 N00037             U1-1      U2-8
     0038 N00038             U2-1      U3-1      U2-9   U1-8
     0039 N00039             U1-12     U5-3
     0040 N00040             U4-9      U3-11     U5-1   U6-6
     0041 N00041             U6-9      U8-8
     0042 N00042             U8-1      U7-5      U8-9   U10-3
     0043 N00043             U10-4     U11-3
     0044 N00044             U9-9      U12-1     U11-1  U10-11
     0045 CIN                U2-10     U3-3      U2-4
     0046 XO                 U4-5      U3-9      U4-1
     0047 SD                 U1-6
     0048 X1                 U5-13     U7-3      U5-9
     0049 X2                 U9-5      U7-13     U9-1
     0050 N00050             U1-3      U5-2      U3-13  U4-12
     0051 X3                 U11-13    U12-9     U11-9
     0052 S1                 U6-3
     0053 N00053             U1-11     U8-10     U7-9   U8-4
     0054 S2                 U6-11
     0055 N00055             U6-8      U11-2     U12-3  U9-12
     0056 YO                 U2-12     U3-5      U4-4
     0057 Y1                 U5-4      U7-1      U5-12
     0058 S3                 U10-8
     0059 Y2                 U8-12     U7-11     U9-4
     0060 Y3                 U11-4     U12-5     U11-12
     0061 COUT               U10-6
     0062 VCC                U1-14     U2-14     U3-14  U4-14
                             U6-14     U5-14     U7-14  U8-14
                             U9-14     U10-14    U12-14 U11-14
     0063 GND                U1-7      U3-7      U2-7   U4-7
                             U6-7      U5-7      U7-7   U8-7
                             U9-7      U10-7     U12-7  U11-7
.PN124
.HE Schematic  Design  Tools                          Hierarchy
```

5.6.7 Invoking The PARTLIST Utility

To obtain a text file part list, use the PARTLIST utility. To execute the PARTLIST utility, enter the following from the DOS command line:

PARTLIST 4BIT.ANN PART.TXT /A <ENTER>PD

PARTLIST is the name of the utility, 4BIT.ANN is the name of the annotation file, PART.TXT is the name of the text file to send the part information (this is useful for examining the results of the utility), /A signifies that the surce is an annotation file, <ENTER> refers to pressing the ENTER> key.

After the utility program has completed, the part

information is sent to the PART.TXT file. To examine the file, use a text editor or word procesing software. Examining the PART.TXT file, results in the following:

```
4BIT  ADDER                    Revised:   October   27,   1986
                               Revision:
Bill  Of Materials   October  29,   1986  1 : 05 : 29  Page  1
```

Item	Quantity	Reference	Part
1	3	U1, U6, U10	74LS32
2	6	U2, U4, U5, U8, U9, U11	74LS08
3	3	U3, U7, U12	74LS04

5.5.8 Invoking The PRINTALL Utility

To obtain a print out on your printer, use the PRINTALL utility. To execute the PRINTALL utility, enter the following from the DOS command line:

PRINTALL 4BIT.ANN /A/S <ENTER>

PRINTALL is the name of the utility, 4BIT.ANN is the name of the annotation file, /A signifies that the surce is an annotation file, /S outputs in scale mode, <ENTER> refers to pressing the <ENTER> key.

Now it is time take a break, while PRINTALL outputs all thirteen worksheets to the printer. Each of the worksheets are ilustrated in the following figures.

(No Figure !)

Figure 5-8. 4BIT Root Worksheet

(No Figure !)

Figure 5-9. FULL ADDER Worksheet

(No Figure !)

Figure 5-10. HALF ADDER Worksheet

(No Figure !)

Figure 5-11. HALF ADDER Worksheet

(No Figure !)

Figure 5-12. FULL ADDER Worksheet

(No Figure !)

Figure 5-13. HALF ADDER worksheet

(No Figure !)

Figure 5-14. HALF ADDER Worksheet

(No Figure !)

Figure 5-15. FULL ADDER Worksheet

(No Figure !)

Figure 5-16. HALF ADDER Worksheet

(No Figure !)

Figure 5-17. HALF ADDER Worksheet

(No Figure !)

Figure 5-18. FULL ADDER Worksheet

(No Figure !)

Figure 5-19. HALF ADDER Worksheet

(No Figure !)

Figure 5-20. HALF ADDER Worksheet

6. U T I L I T I E S

This section describes the utility programs supported by OrCAD/SDT, including:

* ANNOTATE
* BACKANNO
* CLEANUP
* ERCHECK
* NETLIST
* PARTLIST
* PLOTALL
* PRINTALL
* TREELIST

6.1 UTILITY PROGRAM FORMAT NOTATION

The following notation is used to describe the utility programs and the parameters passed to them.

Words shown in CAPITAL letters are program names and are to be entered as shown. When entering, case is not important since DOS converts lowercase letters to uppercase.

Items shown in italics are parameters you supply to the program. For example, the names of source and destination files are parameters.

A switch is a /character followed immediately by a letter. There must be no spaces or other characters between the / and the letter. The letter may be either uppercare or lowercase. Switches may be entered in any order on the invocation line.

.PA

Items in square brackets [] represent parameters which are optional. When you include this optional information, do not include the square brackets when you enter the parameter.

A source path may be one of three different types:

1. The path name to the root sheet of a hierarchical organization of sheets.

2. The path name of the annotation file created by the ANNOTATE program.

3. The path name of a text file which consists of a list of sheet path names. Each sheet path name must be either separated by a space or tab or be on different lines.

The WORKSHEET PREFIX specified in the configuration of OrCAD/SDT will be used as the prefix of the sheet path names in items 1 and 3 above. This prefix may be overridden by placing a drive designation or a back slash at the beginning of the path. For exemple, FILE.SCH will be prefixed by the WORKSHEET PREFIX while A:FILE.CSH and \FILE.SCH will not be prefixed. When overriding, the name must be a valid and complete DOS path name.

Additionally, when overriding a root sheet path name of a hierarchical organization of sheets, only the root sheet path name will be overriden. All sheet scaned will use the WORKSHEET PREFIX and the file name specified for that sheet.

6.2 INVOKING UTILIY PROGRAMS

The OrCAD/SDT utility programs are all post processing, meaning that they are invoked from DOS after you have abandoned from your worksheet editing session.

How you invoke a utility program depends on the type of file structure that you created your worksheet. These file structures are:

- Hierarchy file
- Flat file
- One sheet file

6.2.1 Invoking Utility Programs on Hierarchy File Structures

Invoking a utility programs on hierarchical file structures, enter the name of the utility program followed by the worksheet file name from the DOS command line. For exemple: You have created a hierarchical worksheet with a root file name called COMPUTER.SCH. The hierarchy has three additional hierarchical sheets referenced in MEMORY.SCH. To invoke a utility, just enter following from the DOS command line:

UtilityName computer.sch [optional parameters]

Where UtilityName is the name of the utility program, computer.sch is the name of the root file in the hierarchy , and [optional parameters] are parameters for the utility program such as switch options or other file names.

6.2.2 Invoking Utility Programs On Flat File Structures

Invoking utility programs on flat file structures is handled slightly different than hierarchical files. A flat file structure is a collection of single worksheets. Since the worksheets do not contain references to other worksheet files as a hierarchy does, the flat file strycture must contain a list of the worksheet filename.

The list is a simple text file that you create, that contains the names of the individual worksheets. By invoking a utility program on the name of the text file, all information is merged together into the original worksheets.

To distinguish a flat file, add the /F option switch after the name of the text file to be processed. This informs the utility program that the source is a text file, not a worksheet filename.

For exemple: You have a flat file structure with four file names called SHEET1.SCH, SHEET2.SCH, SHEET3.SCH, and SHEET4.SCH. To run a utility program on all of the file and merge the information back to each of them, place the file names in a text file. If you call the text file BOBSFILE.TXT, enter the following from the DOS command line when invoking a utility program:

Utilityname BOBSFILE.TXT /F [optional parameters]

Where UtilityName is the name of the utility program, BOBSFILE.TXT is the name of the text file, /F is the switch for flat file structure, and [optional parameters] are any parameters that may be added. Such as switch options or destination file names.

6.2.3 Invoking Utility Programs On One Sheet File Structures

Invoking utility program on one sheet file structures is similar to that of a hierarchical structure, except that a /O

option switch is added. The /O treats the worksheet as a single sheet schematic, not a hierarchy.

For exemple: your single worksheet file name is called ONESHEET.SCH. To run a utility program, enter the following on the DOS command line:

UtilityName ONESHEET.SCH /O [optional parameters]

Where UtilityName is the name of the utiliy program, /O treats the worksheet as a one sheet file structure, and [optional parameters] are any parameters that may be added. Such as switch options or destination file names.

6.2.4 Two Floppy Disk System Applications

If you configured your working floppy disks as recommended in Section 2, many of the utility programs will reside on the DRAFT disk. The DRAFT disk should be placed in drive A and the DRIVER/LIBRARY disk in drive B.

When invoking a utility program on a dual floppy system, enter the /E option switch after the file name. The /E switch will display the mesage "Type Any Key To Continue", enabling the system to pause for you to remove the DRIVER/LIBRARY disk in drive B and insert the sheet disk.

For exemple, from the DOS command line enter:

UtilityName FileName /E [optional parameters]

where UtilityName is the name of the utility program, FileName is the name of the your worksheet file, /E enables the system to pause for you to switch disks, and [optional parameters] are any parameters that may be added. Such as switch options or destination file names.

.pa

6.3 ANNOTATE

Purpose:

Scans a hierarchy, flat file, and one sheet file structure n automatically updates all reference designators of library parts that are placed in the worksheet. This includes updating the corresponding pin numbers that are associated with a particular instance of a part with multiple devices in the package.

The results of the program can be placed into either an annotation destination file to be used by other utility programs or can be placed directly into the sheets being scanned using the /M switch.

See Figure 6_1 for an illustration of a worksheet that has

not been run through the ANNOTATE utility program, and Figure 6-2 for a worksheet that has been annotated.

Format:

ANNOTATE source [destination] [/C] [/F] [/M] [/O] [/Q]

Remarks:

The source may be either the root sheet name of hierarchical File structure, the name of a text file in a flat file structure, or the file name of one sheet file structure.

If the source is the name of a text file in a flat file structure, the /F switch must be included. If the source is the file name of a one sheet file structure, the /O switch must be included.

The destination is any valid DOS path name and is where output of the program is to be placed. The ANNOTATE utility program does not process an annotation file. The destination must be included on the commnd line unless the /M switch is specified. The destination path nme is any valid DOS file path name.

ANNOTATE updates reference designatorsin the order they were placed in the worksheet. Any object that has its reference designator manually edited, will be assigned a new reference designator when the worksheet is annotated. If you want to selectively change reference designators, and leave others unmodified, use the BACKANNO utility program.

The/C switch causes the configuration menu to be invoked. This allows the OrCAD/SDT environment to be modified.

The/E switch causes the utility program to display the message "Type Ay Key To Continue", enabling the sistem to pause for you to remove the DRIVER/LIBRARY disk in drive B and insert the SHEET disk. This switch is used only on system with two floppy disk drives.

The/F switch causes the ANNOTATE program to read the source as a text file, for flat file structure applications.

The/M switch causes the annotation information to be merged directly into the source instead of a seaparate annotation file. This switch may be specified for flat file structures as well as hierarchical and one sheet file structures.

If a COMPLEX hierarchy is being merged (refer to Section 5 for a definition of a COMPLEXhierarchy), only the last instance of a multiple referenced sheet will be retained. It is recommended that this switch not be used for COMPLEX hierarchies for this reason. If the/M switch is not specified, then the annotation file will be writen to the destination specified at invocation.

The/O switch causes the file name of the source to be read as a one sheet file structure.

The/Q switch causes the ANNOTATE program to run "quietly".

This means that only the invocation messages and error messages, if any, are displayed. If this switch is not specified, the program will display intermediate tracking activity.
.pn122

6.3.1. Invocation Examples Using Hierarchical Strutured Files

1. To annotate a hierarchical schematic and merge the annotation information into the original sheets:

ANNOTATE root.sch/M

where root.sch is the path and name of the root sheet in the hierarchy and /M merges the information into hierarchical sheets.

2. To annotate a hierarchical schematic and direct the output of the ANNOTATE program to an annotation file:

ANNOTATE root.sch annotation.out

where root.sch is the path and file name to place the annotation information

6.3.2 Flat File Structure Invocation Example

To annotate a flat file structure and merge the annotation information into the original sheets:

ANNOTATE flatfile.txt /F /M

where flatfile.txt is a text file containing a list of schematic file names to be scanned, /F is used to signify that flatfile.txt is a text file, and /M merges the information into flatfile.txt

6.3.3 One Sheet File Structure Invocation Example

To annotate a single sheet schematic and merge the annotation information into the sheet:

ANNOTATE sheetname.sch /O /M

where sheetname.sch is the name of the single sheet schematic, /O is used to signify that sheetname.sch is a single sheet schematic, and /M merges the information into scheetname.sch.

(No Figure !)

Figure 6-1. Unannotated Worksheet

(No Figure !)

Figure 6-2. Annotated Version of Figure 6-1

6.3.4 The Annotation File

He purpose of creating an annotation file is to store the reference designators of a complex hierarchy. A simple hierarchy stbres the reference designators in each of the worksheets. However, a complex hierarchy uses one or more worksheets multiple times. In this case, the annotation file provides a method for storing reference information in binary format.

When using other utility programs on complex hierarchies, the annotation file is used as the source instead of the root schematic. See Section 5, Complex Hierarchies for more information.

6.4 BACKANNO

Purpose:

The BACKANNO programs cans either a hierarchy, flat file, one sheet file structure, or a annotation file and updates part refere,ce designators in your design. The input to the program , a WAS/IS text file containing old and new reference designators, is used to update the schematic worksheets.

Format:

```
BACKANNO source was/is [/A] [/C] [/E] [/F] [/O] [/Q]
```

Remarks:

The source may be either the root sheet name of a hierarchical file structure, the name of a text file ina flat file structure, a annotation file, or the file name of a one sheet file structure.

If thesource is the name of a text file in a flat file structure, the /F switch must be included. If the source is the file name of a one sheet file structure, the /O switch must be included. If the source is the annotation file created by the ANNOTATE program, then the /A switch must be included on the invocation line.

was/is is as text file containing the old and new reference designator pairs. The file may hve any valid DOS path name. The format of the was/is file is discussed later in this Section.

The /A switch causes the BACKANNO program to rea the source path as an annotation file. This will cause the annotation file to be updated.

The /C switch causes the configuration menu to be invoked. This allows the OrCAD/SDT environment to be modified.

The /E switch causes the utility program to display the message "Type Any Key To Continue", enabling the system to pause

for you to remove the DRIVER/LIBRARY disk in drive B and insert the SHEET disk. This switch is used only on systems with two floppy disk drives.

The /F switch causes the BACKANNO program to read the source as a text file, for flat file structure applications.

The /O switch causes the file name of the source to be read as a one sheet file structure.

The /Q switch causes the BACKANNO program to run "quietly". This means that only the invocation messages and error messages if any, are displayed. If this switch is not specified, the program will display intermediate tracking activity.

6.4.1 Invocation Example Using A Hierarchical Structured File

To back annotate sheets in a hierarchical schematic:

BACKANNO root.sch was/is

where root.sch is the path and name of the root sheet in the hierarchy, and was/is is the name of the text file containing the old and new reference designators.

6.4.2 Invocation Example Using A File Structure

The back annotate a flat file structure containing multiple sheets:

BACKANNO flatfile.txt was/is /F

Where flatfile.txt is a text file containing a list of schematic file names to be scanned, was/is is the name of the text file containing the old and new reference designators, and /F is used to signify that flatfile.txt is a text file.

6.4.3 Invocation Example Using A One Sheet File Structure

To back annotate a single sheet schematic:

BACKANNO sheetname.sch was/is /O

Where sheetname.sch is the name of the single sheet schematic, was/is is the name of the file containing the old and new reference designators, and /O is used to signify that sheetname.sch is a single sheet schematic.

6.4.4 Back Annotating Schematics Based on an Annotation File

To back annotate schematics based on the annotation file information:

BACKANNO annotation.out was/is /A

Where annotation.out is the output from the ANNOTATE program, was/is is the name of the text file containing the old and new reference designators, and /A causes the BACKANNO program to read annotation.out as an annotation file.

6.4.5 WAS/IS File Format

The was/is is a simple text file you create, in which you place the old and new reference designators. The file is not required to be delimited by a carriage return after each was/is entry.

The first entry in the text file contains the old reference designator that you want to modify, followed by any number of spaces, tabs, or carriage returns. The next entry in the file is the new reference designator value. The following is a sample of a typical was/is file format:

```
R1    R5
R2    R12
R3    R7
C5    C1
C12   C2
U5C   U1A
U3B   U3A
```

.PA

6.5 CLEANUP

Purpose:

The CLEANUP program scans either a hierarchy, flat file, one sheet file structure, or an annotation file checking for wires, b shs, junctions, labels, module ports, and other objects that are placed on top of each other.

CLEANUP removes duplicate or overlapping wires, b ses, and junctions, and displays warming messages advising you of other duplicate objects.

CLEANUP does not check for objects overlapping part leads, wires overlapping b ses, or wire b s entries overlapping b s bus entries.

This utility program should be used whenever you feel that there may be drawing errors in the worksheet. We recommend that all worksheets be checked withCLEANUP to reduce errors and warnings that may occur when you use other utility programs.

Format:

```
CLEANUP source [destination] [/A] [/C] [/E] [/F] [/O] [/Q] [/R]
```

Remarks:

The source may be either the root sheet name of a hierarchical file structure, the name of a text file in a flat file structure, or the file name of a one sheet of a one sheet file structure.

If the source is the name of a text file in a flat file structure, the /F switch must be included. If the source is the file name of a one sheet file structure, the /O switch must be included. If the source is theannotation file created by the ANNOTATE program, then the /A switch must be included on the invocation line.

The destination is any valid DOS path name and is where the output of the program is to be placed. The destination is an optional parameter and if not specified the output of the CLEANUP program is the console monitor.

If you are processing a worksheet that is very large, the program may return the message: "CLEANUP will need to be repeated for this file". This indicates that CLEANUP has utilized the allocated PC memory for storing temporary worksheet information. If this occurs, repeat the CLEANUP procedure using the /R switch when invoking the program.

The /A switch causes the CLEANUP program to read the source path as an annotation file.

The /C switch causes the configuration menu to be invoked. This allows the OrCAD/SDT environment to be modified.

The /E switch causes the utility program to display the message "Type Any Key To Continue", enabling the system to pause for you to remove the DRIVER/LIBRARY disk in drive B and insert the SHEET disk. this switch is used only on systems with two floppy disk drives.

The /F switch causes the CLEANUP program to read the source as a text file, for flat file structure applications.

The /O switch causes the name of the source to be read as a one sheet file structure.

The /Q switch causes the CLEANUP program to run "quietly". This means that only the invocation messages and error messages if any, are displayed. If this switch is not specified, the program will display intermediate trackin activity.

The /R switch is used to repeat the CLEANUP procedure if a file is too large for one pass. This switch causes theCLEANUP program to repeat the clean up process on a worksheet if needed. The program will repeat a lemited number of times to avoid looping.

6.5.1 Invocation Examples Using Hierarchical Structured Files

1. To check sheets in a hierarchical schematic:

CLEANUP root.sch

Where root.sch is the path and name of the root sheet in the hierarchy.

2. To check a sub-sheet in a hierarchical schematic:

CLEANUP subsheet.sch /O

where subsheet.sch is the path and name of the sub-sheet in the hierarchy, /O signifies that the subsheet.sch file name is a single sheet .

6.5.2 Invocation Examples Using Flat File Structures

1. To check a flat file structure containing multiple sheets:

CLEANUP sheetname.sch /O

where sheetname.sch is the name of the single sheet in the flat file structure, /O is used to signify that sheetname.sch is a single sheey schematic.

6.5.3. Invocation Example Using A One Sheet File Structure

1. To check a single sheet schematic:

CLEANUP sheetname.sch /O

where sheetname.sch is the name of the single sheet schematic, /O is used to signify that sheetname.sch is a single sheet schematic.

6.5.4 Checking Schematics Based on a Annotation File

1. To check schematics based on the annotation file information:

CLEANUP annotation.out /A

Where annotation.out is the output from the ANNOTATE program, /A causes the CLEANUP program to read annotation.out as an annotation file.

.pa

6.6 ERCHECK

Purpose:

Performs a basic electrical rules check on a group of schematicsheets that are specified as a source input. The ERCHECK program will scan either a hierarchy, flat file, one sheet file structure, or the annotation file output from the ANNOTATE program. The sheets are checked for unused inputs on parts, unlabeled wires connected to a b s, and invalid connections such as two part pins defined as outputs wired together.

Format:

ERCHECK source [destination] [/A] [/C] [/E] [/F] [/O] [/Q]

Remarks:

The source may be either the root sheet name of a hierarchical file structure, the name of a text file in a flat file structure, or the file name of a one sheet file structure.

If the source is the name of a text file in a flat file structure, the /F switch must be be included. If the source is the file name of a one sheet file structure, the /O switch must be included. If the source is the annotation file created by the ANNOTATE program, then the /A switch must be included on the invocation line.

The destination is any valid DOS path name and is where the output of the program is to be placed. The destination is an optional parameter and if not specified the output of the ERCHERCK program is the console monitor.

The /A switch causes the ERCHECK program to read the source path as an annotation file.

The /C switch causes the configuration menu to be invoked. This allows the OrCAD/STD environment to be modified.
.PN130

The /E switch causes the utility program to display the message "Type Any Key To Continue", enabling the system to pause for you to remove the DRIVER/LIBRARY disk in drive B and insert the SHEET disk. This swich is used only on systems with two floppy disk drives.

The /F switch causes the ERCHECK program to read the source as a text file, for flat file structure applications.

The /O switch causes the file name of the source to be read as a one sheet file structure.

The /Q switch causes the ERCHECK program to run "quietly". This means that only the invocation messages and error messages if any, are displayed. If this witch is no specified, the program will display intermediate tracking activity.

6.6.1. Invocation Examples Using Hierarchical Structured Files

1. To check sheets in a hierarchical schematic:

ERCHECK root.sch

Where root.sch is the path and name of the root sheet in the hierarchy.

2. To check a sub-sheet in a hierarchical schematic:

   ERCHECK subsheet.sch /O

   Where subsheet.sch is the path and name of the sub-sheet in the hierarchy. /O signifies that the subsheet.sch file name is a single sheet.

3. To check a hierachical schematic and direct the output of the ERCHECK progam to a file:

   ERCHECK root.sch whatfile

   Where root.sch is the path and name of the root sheet in the hierarchy and whatfile is the path and filename to place the ERCHECK information.

6.6.2. Invocation Examples Using Flat File Structures

1. To check a flat file structure containing multiple sheets:

   ERCHECK flatfile.txt/F

   Where flatfile.txt is a text file containing a list of schematic file names to be scanned. /F is used to signify that flatfile.txt is a text file.
2. To check one sheet in a flat file structure:

   ERCHECK sheetname.sch/O

   Where sheetname.sch is the name of the single sheet in the flat file structure. /O is uset to signify that sheetname.sch is a single sheet schematic.

3. To check a flat file structure and direct the output of the ERCHECK program to a file:

   ERCHECK flatfile.txt whatfile /F

   Where flatfile.txt is a text file containing a list of schematic file names to be checked, whatfile is the path and file name to place the ERCHECK information. /F is used to signify that flatfile.txt is a text file.

6.6.3. Invocation Examples Using A One Sheet File Structure

1. To check a single sheet schematic:

   ERCHECK sheetname.sch /O

   Where sheetname.sch is the name of the single sheet schematic. /O is used to signify that sheetname.sch is a

single sheet schematic.

2. To check a single schematic and direct the output of the ERCHECK program to a file:

   ERCHECK sheetname.sch whatfile /O

   Where sheetname.sch is the name of the single sheet schematic. whatfile is the path and file name to place the ERCHECK information, and /O is used to signify that sheetname.sch is a single sheet schematic.

6.6.4. Checking Schematics based on a Annotation File

1. To check schematics based on the annotation file information:

   ERCHECK annotation.out /A

   Where annotation.out is the output from the ANNOTATE program. /A causes the ERCHECK program to read annotation.out as an annotation file.

6.6.5. Typical ERCHECK Messages and Resolutions

Listed below, are the most common error messages produced by ERCHECK and possible solutions to resolve the errors.

Message

```
<<<WARNING>>> Unconnected MODULE PORT "......." at X= .......
              at Y= ........
```

Check For:

A bus that is not properly labeled as internal. It must be named in the form: BUSNAME[0..n]. Any module port that is connected to a bus must also be named in the proper from: BUSNAME[0..n]. For further information, see Sections 6.7.7 through 6.7.9.

Message:

WARNING - POWER Supplies are CONNECTED .... <-> .....

Check For:

This may be a warning that is acceptable in your design. If you intentionally connected two power supplies together, this warning will appear. If you did not connect two power supplies together, this indicates that a potential problem may exist.

Message:

WARNING - INPUT has NO Driving Source ....

Check For:

Again, this warning may be acceptable in your design. If you intentionally did not connect wires to the input pins of a library part this warning will appear.

If wires are connected to an input pin and this message appears, you may have two wire ends overlapping or a wire overlapping a part pin. Wires must be connected end-to-end. Run the CLEANUP utility, and check for wire ends overlapping part pins (CLEANUP does not remove these).

Message:

<<<ERROR>>> Module Port on a bus does not have a proper

Check for:

When a module port is connected to a bus, it must be in the format:

BUSNAME[0..n]. For further information, see Section 6.7.9.

Message:

<<<ERROR>>> Bus Label does not have a proper format ... can not process.

Check for:

When a bus label is connected to a bus, it must be in the format:

BUSNAME[0..n]. It must also be an internal label type. For further information, see Section 6.7.7.

Message:

<<<ERROR>>> Sheet Net on a bus does not have a proper format .. can not process ..

Check for:

This error typically results from a bus connected to a hierarchical sheet net. The net name match the name of the bus that it is connected to. The form should be: BUSNAME[0..n]. For additional information, see Section 5.1 and 5.4.

.pa

6.7. NETLIST

Purpose:

The NETLIST program scans either a hierarchy, flat file, one sheet file structure, or a annotation file and generates a net list in a number of selectable formats.

Format:

NETLIST source [destination] [format] [/A] [/C] [/E] [/F] [/H] [/M] [/N] [/O] [/Q] [/S] [/P]

Remarks:

The source may be either the root sheet name of a hierarchical File structure, the name of a text file in a flat file structure, or the file name of a one sheet file structure.

If the source is the name of a text file in a flat file structure, the /F switch must be included. If the source is the file name of a one sheet file structure, the /O switch must be included. If the source is the annotation file created by the ANNOTATION program, then the /A switch must be included on the invocation line.

The destination is any valid DOS path name and is where the output of the program is to be placed. The destination is an optional parameter and if not specified the output of the NETLIST program is the console monitor.

While NETLIST is processing worksheets, a sequence of asterisks (*) and periods (,) are displayed on the screen. This informs you that processing is currently taking place.

The format is the name of the special file format that you want the net list generated in. If you do not specify a format, then the format defaults to EDIF. The format may be any one of the following:

APPLICONBRAVO
APPLICONLEAP
ALGOREX
CALAY |
CADNETIX
COMPUTERVISION
EDIF
FUTURENET ~
INTERGRAPH
MULTIWIRE
PCAD
SPICE /\
SCICARDS
RACALREDAC |
TELESIS
VECTRON *
WIRELIST
| Creates two output components ans nets
/\ Creates two output files nets and net map
- PINLIST or NETLIST

The /A switch causes the NETLIST program to read the source path as an annotation file.

The /C switch causes the confuguration menu to be invoked. This allows the OrCAD/SDT environment be modified.

The /E switch causes the utility program to display the message "Type Any Key To Continue", enabling the system to pause for you to remove the DRIVER/LIBRARY disk in drive B and insert the SHEET disk. This switch is used only on systems with two floppy disk drivers.

The /F switch causes the NETLIST program to read the source as a next file, for flat file structure applications.

The /H switch removes all duplicate sheets in a complex hierarchy. This switch is only used when generating a EDIF net list output. The resultant net list will be completely hierarchical.

The /M switch causes the signal type of labeled or module port signals to be specified in the Future Net PINLIST format only. This is used by many gate array and wirewrap vendors.

The /N switch outputs a NETLIST instead of a PINLIST. This switch is only used when generating a FUTURENET net list output.

The /O switch causes the file name of the source to be read as a one sheet file structure.

The /P switch displays pin numbers instead of pin names in the net list output. This switch may be used on all formats.

The /Q switch causes the NETLIST program to run "quietly". This means that only the invocation messages and error messages if any, are displayed. If this switch is not specified, the program will display intermediate tracking activity.

The /S switch is used to generate one of the special netlist formats. If the switch is not specified, the net list is generated in EDIF format.

Rules:

If you are using the NETLIST utility program, certain rules must be observed when you create your schematic worksheets. In particular, you must properly label buses, bus members, signals, ans module ports. In addition, power must also be properly handled. Handling each of these items is discussed further in this NETLIST Section.

In addition, if wires, buses, or library objects are placed end-to-end in the worksheet, the NETLIST program interprets this as a physical connection. This is true if a junction is not placed at the meeting point.

6.7.1. Invocation Examples Using Hierarchical Structured File

1. To obtain a net list of a hierarchical schematic:

NETLIST root.sch

Where root.sch is the path and name of the root sheet in the hierarchy.

2. To obtain a net list of a sub-sheet in a hierarchical schematic:

NETLIST subsheet.sch /O

Where subsheet.sch is the path and name of the sub-sheet in the hierarchy. /O signifies that the subsheet.sch file name is a single sheet.

3. To obtain a net list of a hierarchical schematic and direct the output of the NETLIST program to a file:

NETLIST root.sch whatfile

Where root.sch is the path and name of the root sheet in the hierarchy and whatfile is the path and file name to place the NETLIST information.

4. To obtain a netlist of a hierarchical schematic in CALAY format and direct the output of the NETLIST program to a file:

NETLIST root.sch whatfile CALAY /S

Where root.sch is the path and name of the root sheet in the hierarchy, whatfile is the path and file name to place the NETLIST information, CALAY is the desires format, /S signifies that a special format is desired (Calay in this example).

6.7.2. Invocation Examples Using Flat File Structures

1. To obtain a net list of a flat file structure containing multiple sheets:

NETLIST flatfile.txt /F

Where flatfile.yxy is a text file containing a list of schematic file names to be scanned. /F is used to signify that flatfile.txt is a text file.

2. To obtain a net list of one sheet in a flat file structure:

NETLIST sheetname.sch /O

Where sheetname.sch is the name of the single sheet in the flat file structure. /O is used to signify that sheetname.sch is a single sheet schematic.

.pa

3. To obtain a net list of a flat file structure and direct the output of the NETLIST program to a file:

NETLIST flatfile.txt whatfile /F

Where flatfile.txt is a text file containing a list of schematic file names to be checked, whatfile is the path and file name to place the NETLIST information. /F is used to signify that flatfile.txt is a text file.

4. To obtain a net list of a flat file structure in RACAL-REDAC format and direct the output of the NETLIST program to a file:

NETLIST flatfile.txt whatfile RACALREDAC /F/S

Where flatfile.txt is a text file containing a list of schematic file names to be checked, whatfile is the path and file name to place the NETLIST information. RACALREDAC is the desired format. /F is used to signify that flatfile.txt is a text file. /S signifies that a special format is desired (Racal-Redac in this example).

6.7.3. Invocation Examples Using A One Sheet File Structure

1. To obtain a net list of a single sheet schematic:

NETLIST sheetname.sch /O

Where sheetname.sch is the name of the single sheet schematic. /O is used to signify that sheetname.sch is a single sheet schematic.

2. To obtain a net list of a single schematic and direct the output of the NETLIST program to a file:

NETLIST sheetname.sch whatfile /O

Where sheetname.sch is the name of the single sheet schematic, whatfile is the path and filename to place the NETLIST information, and /O is used to signify that sheetname.sch is a single sheet schematic.

3. To obtain a net list of a single schematic in SCICARDS format and direct the output of the NETLIST program to a file:

NETLIST sheetname.sch whatfile SCICARDS /O/S

Where sheetname.sch is the name of the single sheet schematic, whatfile is the path and file name to place the NETLIST information. SCICARDS is the desired format, /O is used to signify that sheetname.sch is a single sheet schematic, /S signifies that a special format is desired (SCICARDS in this example).

6.7.4. Net Lists Based on an Annotation File

1. To obtain a net list based on the annotation file information:

NETLIST annotation.out /A

Where annotation.out is the output from the ANNOTATE program, /A causes the NETLIST program to read annotation.out as an annotation file.

6.7.5. The Limits of NETLIST

This size of a netlist is limited by the available system memory. As the net list program runs, it processes each schematic file separately. The limit with 640k of system memory is 8.000 wire segments and about 2,000 to 10,000 device pins per schematic file. This includes only those objects on a single sheet.

For each sheet, if the EDIF format has been selected, the net list is output and resolution of signals is not required. EDIF contains all of the information that is needed. If another format has been selected, then the NETLIST program saves information on any signal that has not been resolved (signals that go off the sheet) separately from those that are complete nets. This process is completed for each sheet in the set of schematic files presented to the net lister.

After all of the sheets have been processed, the previously unresolved nets are combined to complete the connection of signals across sheet boundaries. In this phase, there is a limit of about 1,000 signal names and a limit of around 1,000 to 6,000 total nets. For a flat file structure, this phase combines all sheets at the same time. If there is not enough memory for this phase to complete, then the design must be made hierarchical.

For a hierarchical design, an incremental resolution is made and the limits outlined above apply to each pass of the resolution process. This means that a much larger design can be handled with the hierarchical structure than the flat file structure.

6.7.6. Notes on Particular Formats

Many netlist formats require you to follow special pin numbering conventions. For example, OrCAD libraries have some of the pin names as text. These may need to be converted to whole numbers, depending on the desired netlist format. To ceonvert pin names to whole numbers, see the procedures described in the SPICE netlist.
.pa

FUTURENET

The FutureNet system has to connectivity output formats, PINLIST and NETLIST. The PINLIST format is a part oriented list that lists for each pin of a part, the net to which the pin is attached. The NETLIST is a net oriented net list that lists for each net, the part pins that are attached. The NETLIST format is extensively sorted by FutureNet. OrCAD does NOT do all of this sorting.

The FutureNet output is limited by the amount of available system memory. This is due to the requirement than the parts and node need to be cross referenced. The limit is approximately 8,000 parts and 12,000 total nets. This should be enough for most designs. If you are doing gate array designs and need additional capacity, it may be neccesary to partition the design. Contact OrCAD or your ASIC vendor for additional information. It is recommended that at least 512K memory be installed in your PC.

PCAD

OrCAD uses a subset of the full PDIF specification. As detailed in the PDIF Specification documentation, a NETLIST IMPORTATION SPECIFICATION is used to transfer a PCB net list into the PCAD system. The PCAD system is a part oriented net list and is limited by the amount of available system memory. This is due to the requirement that the parts and nodes need to be cross referenced.

The limit is approximately 8,000 parts and 12,000 total nets. This should be enough for most designs. If you are doing gate array designs and need additional capacity, it may be necessary to partition the design. Contact OrCAD or your ASIC vendor for additional information. It recommended that at least 512K memory be installed in your PC.

Part values are suffixed with .PRT and used as the instance filename.

The following text contains two examples of transporting a netlist generated using the OrCAD/SDT NETLIST in PCAD format. Example one shows the ENVIRONMENT statement of the OrCAD netlis can be modified to link the layering file necessary for the PCAD PCB layout software. Because some versions of the PCAD utility software will not link the OrCAD/SDT PCAD compatible netlist with the BINARY layering file a second example given showing how the OrCAD/SDT PCAD compatible ASCII netlist file can be merged with a ASCII layering file.

Example One: Changing the netlist "ENVIRONMENT" statement

The sample net list below is an example of the standard header line generated by OrCAD's NETLIST utility. In order for an OrCAD/SDT PCAD netlist to be compatible to the PCAD PCB layout software the header line must be modified to include a PCAD BINARY layering file such as the generic "LAYS.PCB".

```
.pn141
.he Schematic Design Tools                                NETLIST

(COMPONENT OrCAD.NET   (ENVIRONMENT OrCAD.SDT)  (DETAIL SUBCOMP (I
2114.PRI U6 (CN
     1 ADDRESS6
     2 ADDRESS5
     3 ADDRESS4
     4 ADDRESS3
     5 ADDRESS0
     6 ADDRESS1
     7 ADDRESS2
     8 XN00002
     9 BSS
    10 WR*/RD_1
    11 D3_1
    12 D2_1
    13 D1_1
    14 D0_1
    15 ADDRESS9
    16 ADDRESS8
    17 ADDRESS7
    18 VDD
))
)))
```

The net list below contains a modified header line to which the ENVIRONMENT statement has been modified to replace "OrCAD.SDT" with the BINARY layering file "LAYS.PCB" from PCAD.

```
(COMPONENT OrCAD.NET   (ENVIRONMENT LAYS.PCB)  (DETAIL SUBCOMP
(I 2114.PRI U6 (CN
     1 ADDRESS6
     2 ADDRESS5
     3 ADDRESS4
     4 ADDRESS3
     5 ADDRESS0
     6 ADDRESS1
     7 ADDRESS2
     8 XN00002
     9 VSS
    10 WR*/RD_1
    11 D3_1
    12 D2_1
    13 D1_1
    14 D0_1
    15 ADDRESS9
    16 ADDRESS8
    17 ADDRESS7
    18 VDD
))
)))
```

Example Two: Merging PCAD netlist with ASCII layerinf file

The net list below contains an example of a layering file that has been converted from a BINARY file format to an ASCII file format using the PDIF-OUT utility available with PCAD.

```
(COMPONENT LAYS.PCB

     (ENVIRONMENT
     (PDIFrev 1.30)
     (Program "PC-CARDS Version 0.02")
     (DBtype "PC-Board")
     (DBrev  1.00)
     (DBtime "Oct, 22, 1985        2.03 p.m.      ")
     (DBunit "MIL")
     (DBgrid 1)
    (lyrstr "PADCOM" 7 "FLCOMP" 7 "PADSLD" 8 "FLSOLD" 8 "PADINT" 9
            "FDINT" 9 "GNDCON" 10 "FIGCON" 10 "GNDCLR" 12
            "FLGCLR"  12  "PWRCLR"  13 "FLPCLR"  13  "SLDMSK"  14
            "FLSMSK" 14 "DRILL" 15 "FLDRLL" 15 "PIN" 4 "BRDOUT" 4
            "FLTARG" 4 "SLKSCR" 6 "DEVICE" 5 "ATTR" 6 "REFDES" 6
            "COMP" 1 "SOLDER" 2 "INT1" 3)
)
(USER
 (PCAD
  (Vw 750 360 8)

(lv 24 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 0 0 2 2 0)
(Gs 50 50)
)
)
(DISPLAY
 (Ly "COMP")
```

```
 (Ls "SOLID") (Wd 15)
 (Ts 125) (Tj "CC") (Tr 11) (Tm "N")
       )

(SYMBOL
 (PIN_DEF
 )

(PIC
)

(ATR
 (IN
   (Orq -32767 -32767)
   (Ty 255)
  )
 )
)

 (DETAIL
)

 (NET_DEF
)

 (PAD_STACK
)

  (SUBCOMP
  )
 )
)
```

Illustrated below is an example of the modifications that are needed to allow the OrCAD PCAD netlist to be merged with the ASCII layering file. Note that the header line of the OrCAD PCAD netlist has been deleted as well as the three (3) brackets of the netlist file.

```
(I 2114.PRT U6 (CN
    1 ADDRESS6
    2 ADDRESS5
    3 ADDRESS4
    4 ADDRESS3
    5 ADDRESS0
    6 ADDRESS1
    7 ADDRESS2
    8 XN00002
    9 VSS
   10 WR*/RD_1
   11 D3_1
   12 D2_1
   13 D1_1
   14 D0_1
   15 ADDRESS9
   16 ADDRESS8
   17 ADDRESS7
   18 VDD
))
```

Finally, ilustrated below is an example of an OrCAD PCAD netlist that has been merged into an ASCII layering file. After the merging process is finished the resultant file can be converted back into a BINARY formated file by using the PDIF-IN utility available from PCAD. The resulting file from the PDIF-IN utility is ready for use with the PCAD PCB layout software. It should be noted here that the file name given is the COMPONENT statement of the ASCII layering file should be changed to avoid the possibility of overwriting the existing BINARY layering file "LAYS.PCB".

```
(COMPONENT LAYS.PCB

 (ENVIRONMENT
  (PDIFrev 1.30)
  (Program "PC-CARDS Version 0.02")
  (DBtype "PC-Board")
  (DBrev  1.00)
  (DBtime "Oct, 22, 1985        2.03 p.m.      ")
  (DBunit "MIL")
  (DBgrid 1)
  (lyrstr "PADCOM" 7 "FLCOMP" 7 "PADSLD" 8 "FLSOLD" 8 "PADINT" 9
            "FDINT" 9 "GNDCON" 10 "FIGCON" 10 "GNDCLR" 12
            "FLGCLR"  12  "PWRCLR"  13 "FLPCLR"  13  "SLDMSK"  14
            "FLSMSK" 14 "DRILL" 15 "FLDRLL" 15 "PIN" 4 "BRDOUT" 4
            "FLTARG" 4 "SLKSCR" 6 "DEVICE" 5 "ATTR" 6 "REFDES" 6
            "COMP" 1 "SOLDER" 2 "INT1" 3)
)
(USER
 (PCAD
  (Vw 750 360 8)

(lv 24 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 0 0 2 2 0)
(Gs 50 50)
)
)
(DISPLAY
 (Ly "COMP")
 (Ls "SOLID") (Wd 15)
 (Ts 125) (Tj "CC") (Tr 11) (Tm "N")
       )
(SYMBOL
 (PIN_DEF
 )

(PIC
)

(ATR
 (IN
   (Orq -32767 -32767)
   (Ty 255)
  )
 )
)

 (DETAIL
)
```

```
 (NET_DEF
)

 (PAD_STACK
)

  (SUBCOMP
(I 2114.PRT U6 (CN
    1 ADDRESS6
    2 ADDRESS5
    3 ADDRESS4
    4 ADDRESS3
    5 ADDRESS0
    6 ADDRESS1
    7 ADDRESS2
    8 XN00002
    9 VSS
   10 WR*/RD_1
   11 D3_1
   12 D2_1
   13 D1_1
   14 D0_1
   15 ADDRESS9
   16 ADDRESS8
   17 ADDRESS7
   18 VDD
  ))
  )
 )
)
```

SPICE:

OrCAD can create a net list that is larger the most PC based Spice programs will accept. Consult your Spice manual for the limits. If your PC meets the memory requirements of Spice, the largest Spice net list is capable of being generated. This should work for all PC versions of Spice.

The part value is used to pass modeling information to the net list. A special Spice library should be created to support the proper model pin numbers. To implement this, use the DECOMPLEXE library utility and convert the desired OrCAD supplied libraries to a source file (refer to Section 7). Make the appropriate changes to the source file and recompile the modified library back to an object file using COMPOSER.EXE.

All library part pin names should be changed to reflect the model note index. For example, a resistor has two nodes. The library definition of the resistor (the bitmap has been omitted for clarity only) is outlined as follows:

```
'RESISTOR'
REFERENCE 'R'
(X Size =) 2   (Y Size =) 3        (Parts per Package =) 0
T1        SHORT     PAS '1'
B1        SHORT     PAS '2'
               .
               .
               .
```

A bipolar tranzistor would have the pin names listed as: (1) collector, (2) base, and (3) emitter. The library definition of the tranzistor outlined as follows:

```
'NPN'
REFERENCE      'Q'
(X Size =)     2    (Y Size =)     2    (Parts per Package
=)        0
L1      SHORT     IN   '2'
B2      SHORT     IN   '3'
T2      SHORT     IN   '1'
(  0).........##..........#
(  1).........##.........#.
(  2).........##........#..
(  3).........##.......#...
(  4).........##......#....
(  5).........##.....#.....
(  6).........##....#......
(  7).........##...#.......
(  8).........##..#........
(  9).........##.#.........
( 10)###########...........
( 11).........##.#.........
( 12).........##..#........
( 13).........##...#.......
( 14).........##....#.....#
( 15).........##.....#...##
( 16).........##......#.###
( 17).........##.......####
( 18).........##......#####
( 19).........##.....######
( 20).........##....#######
```

The node numbers created by OrCAD are placed in a MAP file so that you may reference the Spice node numbers with the node names that you have specified in your schematic.

CALAY, RACALREDAC, and VECTRON:

The NETLIST program generates an additional component file when you select the CALAY, RACALREDAC, and VECTRON formats. CALAY and RACALREDAC generate a .CMP file extension and VECTRON generates a .PLI file extension.

.pa

6.7.7. Applications of Internal Labels

Internal labels are used to connect signals together from one worksheet area to another without using wires or buses. They are also mandatory for labeling buses when they are placed in the worksheet. As a rule, you may place any number of internal labels on a bus or a wire.

For example, you have a signal labeled ABC in the worksheet and you would like to connect another object in the worksheet to the same signal. Instead of drawing a wire from ABC on one side of the worksheet to the other object, you can label each signal with a internal label as Figure 6-3 illustrates below:

U1A

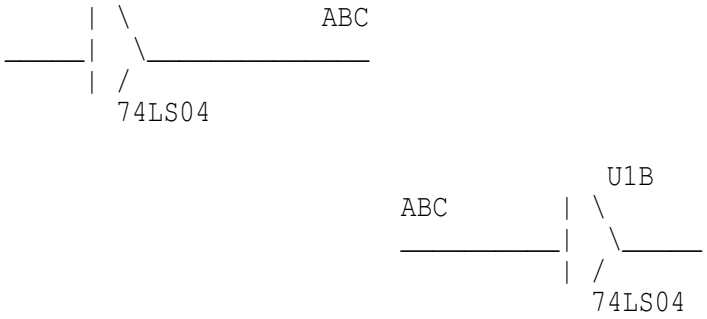


Figure 6-3. An Example of Using an Internal Label

The other use of an internal label is for labeling buses. Every bus must be labeled for the NETLIST program to properly associate the bus with the individual members of the bus.

Internal bus labels must be in the form:

BUSNAME[0,n]

Where BUSNAME is called the "prefix" and it represents the name of the bus. [0,n] is called the "suffix", where n represents the decimal number of the last member of the bus: only a zero (0) is valid in first portion of the suffix ([2,n] for example, is not valid). The prefix and suffix must have no space between them.

Typical examples would be:

```
ADDR[0..31]    (This bus has 32 members)
DATA[0..15]    (This bus has 16 members)
CONTROL[0..3]  (This bus has  4 members)
A[0..90]       (This bus has 91 members)
```

.pa

Figure 6-4 shows an example of properly labeling a bus with an internal label.

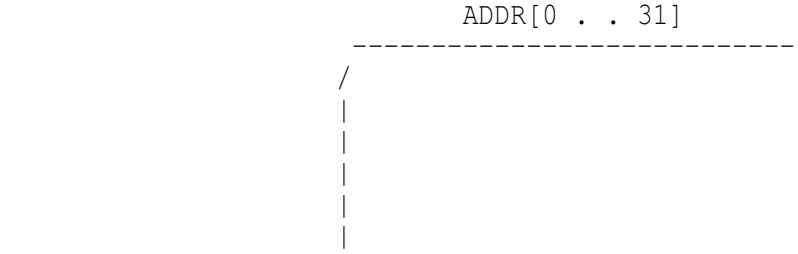


Figure 6-4. Properly Labeling a Bus

6.7.8. Applications of Bus Member Labels

Bus member labels are used to label the individual members that emanate from a bus. As a rule, you may place any number of bus member labels on a bus member. Since bus member labels are associated with a bus, they must be labeled in a form that corresponds to the bus they came from. The form is:

BUSNAMEx

Where BUSNAME is the same prefix name as the internal bus label, x is a decimal number in the range of [0 - n] yaken from the suffix of the internal bus label, and n represents the decimal number of the last member of the bus. BUSNAME and x must have no space between them.

An example of the use of internal and bus member labels is illustrated in Figure 6-5 below. Label A[0,9] is placed on the bus and is a internal label. Labels A0 through A9 are bus member labels, that identify each number of the bus. Notice that the letter "A" in each of the bus member labels matches the prefix of the internal label on the bus, which is also the letter "A".

(No Figure !)

Figure 6-5. Application of an Internal and Bus Member Label

6.7.9. Applications of Module Ports

Any signal that is go off the worksheet must does so via a module port. In a flat file strcture, all module ports are considered to be global. In a hierarchical structure, the signals that connect to tke level above a particular sheet do so via module ports. These module ports may be either, single signal or a bus.

In a hierarchy, the module port must have the exact same name as the sheet net name to wich it connects in the sheet symbol. If the signal is a bus, then both the module port and the sheet net name must be of the bus identification format. This format is:

BUSNAME[0..n]

Where BUSNAME is called the "prefix" and it represents the name of the bus. [0..n] is called the "suffix", where n represents the decimal number of the last member of the bus. The prefix and suffix muist have no space between them.

Typical examples would be:

ADDR[0..31]
DATA[0..15]
CONTROL[0..3]
A[0..90]

When a bus is connected to a module port, in both flat file and hierarchical file structures, the suffix of the internal label on the bus must be identical to the suffix of the module port name. The prefix of the module port and the internal label on the bus may be different. Figure 6-6 illustrates three examples of a properly connecting buses to module ports. Notice that the internal bus label suffix, [0..15], is identical to the module port suffix.

(No Figure !)

Figure 6-6. Connecting Buses to Module Ports

Signals (individual wires) may be connected directly to module ports. The signal does not have to be labeled unless it connects to a bus. In thsi case, you must specify which bus member is connected to the module port. Module ports that are connected to signals may be named in any format. They are not required to have a suffix. Figure 6-7 illustrates typical examples of signals that are connected to module ports.

6.7.10. Splitting Buses

A special feature of DRAFT, enables you to split buses in your worksheet. Figure 6-8 illustrates an example of how to do this. In this figure, the bus labeled A/D[0..15] is an internal label attached to a module port having the same name. Members of this bus are connected to U7 through bus member labels: A/D0 through A/D15.

(No Figure !)

Figure 6-8. Splitting a Bus

.pn150

To split this bus, place two new buses anywhere in the worksheet and label them as follows. One bus is named AL[0..7] and is entered as an internal label. Signals to the bus are transferred via bus member labels: A/D0 through A/D7 which are "implied" connections to any other bus member label having the same names. Labels AL0 through AL7, are also bus member labels. They perform the actual splitting and transfer the signals to the bus AL[0..7].

The other bus, named AH[0..7], is labeled as an internal label. Bus member labels A/D8 through A/D15 make implied connections to similar label names. Bus member labels AH0 through AH7 split the higher address line from U7 to the bus labeled AH[0..7].

When splitting buses, do not physically connect the split bus members to the main bus. In this example, do not connect bus member labels A/D0 through A/D15 to the main bus labeled A/D[0..15].

6.7.11. Multiple Labels On a Bus

As previously mentioned, a bus may have more than one internal label placed to it. Figure 6-9 illustrates a bus contains two internal labels: A[0..3] and B[0..3].

Bus member labels A0 through A3 are associated with internal label A[0..3] and bus members label B0 through b3 are associated with internal label B[0..3]. The Figure shows that the internal labels are placed some distance away from each other. In actual applications, they may be placed anywhere on the bus and still be

associated with their respective memeber labels.

(No Figure !)

Figure 6-9. Multiple Label on a Bus

6.7.12. Combining Internal and Bus Member Labels

Figure 6-10 illustrates an example of combining internal and bus member labels. Internal label MEM[0..11] is placed on the bus which contains 12 members. U1 is connected to the bus via bus member labels MEM0 through MEM11.

Notice that bus member label MEM10 has a label C\S\ placed next to it, and bus member label MEM11 has a label W\E\ placed next to it. C\S\ and W\E\ are internal labels that have been placed on the bus members. These internal labels connect C\S\ and W\E\ to U2 and U3. In the case of U3, C\S\ and W\E\ are connected to the 2114 device without being physically connected to the bus.

Bus member labels MEM0 through MEM9 on U2 and U3 connect the 2114 address lines (A0 - A9) to the bus.

(No Figure !)

Figure 6-10. Combining internal and bus member labels

6.7.13. Connectting Bus Labels to Module Ports

Figure 6-11 illustates connecting bus labels to module ports. The main bus is labeled with an internal label in the form: BUS[0..10]. U1 is connected to the bus via bus member labels BUS0 through BUS10.

Signal W\E\ is transferred off the worksheet via module port W\E\. Since the signal is transferred through bus BUS[0..10], the signal must be labeled with BUS10. BUS10 ia a bus member label.

The R\A\S\ signal is labeled with a internal label to be used elsewhere in the worksheet. Since the R\A\S\ signal is transferred through the bus, it must also be labeled with a bus member label BUS8. This associates the signal with R\A\S\ on U1.

Finally, bus BUS[0..10] also leaves the worksheet. It is connected to a module port having the same name.

(No Figure !)

Figure 6-11. Connecting Labels in Module Ports

6.7.14. Processing Module Ports and Internal Labels

Whenever a bus module port is processed by the NETLIST utility, the bus module port will be split into bus member module ports. The names assigned are the same as would be used when

placing bus member labels on a schematic. For example:

A[0..3] would become:
A0
A1
A2
A3

If you have named any module ports with the same name as one of these split: bus module port names, they will be connected together.

Internal labels will be made part of the net list when ever possible. In the final output, the internal label will be suffixed with a delimiter (usually _ but will be different if the net list format selected does not allow _).

If you do not have unique sheet numbers in the title block for all sheets, the net list may contain duplicate internal label references. No check is made for the uniqueness of sheet numbers. Additionally, if you have other labels that are in the same form as these internal labels, such as in the following example, the net list will not be checked for duplicates.

Read_Write_Bar (internal label on sheet 5)
would become: Read_Write_Bar_5

AD0 (bus member label of internal bus on sheet 5)
would become: AD0_5

6.7.15. Handling Power

Power connections are handled in a number of ways. Most parts in the libraries supplied by OrCAD have defined power and ground pins. These pins are hidden from the drawing, but never the less are part of the symbol definition. These hidden pins have special meaning when creating a net list.

To make connection to these power pins, a power object is used (PLACE Power command). The net list program will connect the hidden power pins to a power object that has been placed in the worksheet having the same name as the parts supply pin.

A power pin in a library part is isolated in the worksheet that the part is placed in. To get the power pin connected off the worksheet, you have to define how it is done: it is not done automatically. The power pin will be global if a power object is placed in the worksheet. Global means that a signal (power in this case) connects to other signals that are global having the same name. This is true for any worksheet file structure. As you will see later, when a power object is connected to a module port, power becomes local in the worksheet.

For example, in Figure 6-12, you have placed two TTL devices in the worksheet. Both devices have been defined to have a VCC and a GND power pin in the library source file. To make the connection to the VCC pin in the net list, a power object named VCC must be placed in the worksheet. It does not matter where in the sheet this power object is placed, nor does it matter how many devices there are to connect to the power object.

(No Figure !)

Figure 6-12. Placing a VCC Power Object in the Worksheet

If an object has a power pin defined as VDD in the library source file, then the power object placed in the worksheet must also be named VDD. Whenever the worksheet contains objects with different types of power pins, then a power object must be placed in the worksheet that has been named identically to a parts power pin.

The power object may be connected to power objects that have the same name, or power objects with different names or labels. Any number of power objects may be placed on the sheet at a time, but a minimum of one is required in each worksheet for the NETLIST program to properly connect the hidden power pins.

6.7.16. Connecting Different Power Object Names

Invoke OrCAD supplied libraries, many of the devices have been dermed to have a VCC as the positive supply voltage pin. Others have VDD defined as the positive supply voltage pin. If you want them both to operate from the same power supply, you must connect them together.

Likewise, many of the libraries have GNF and VSS defined as the return power pins. If you want to have them connected together and be common to each other, you must connect them connect together also.

To connect power supply pins together, or connect a power supply pin to any other supply voltage, you must place a power object for each different supply in the worksheet. One power object must be named the same as one of the supply voltages, VDD for example. The other power object must be named the same as the remaining supply voltage, VCC for example. Finally, the power objects that are placed in the worksheet must be connected together with a wire. Figure 6-13 illustrates four examples.

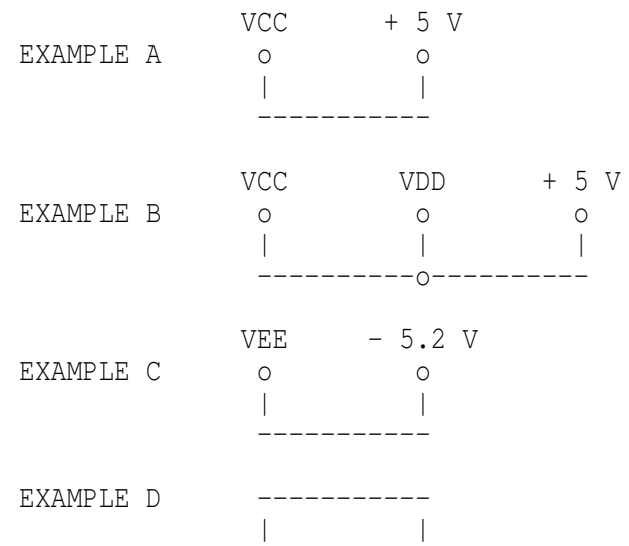


Figure 6-13. Power Supply Connections

Example A illustrates a power object named VCC in connected to a + 5 volt power supply. In the net list output, every object that has VCC pin will be connected to a +5 volt power supply. This assumes that your design contains a power supply with a power object named +5V attached. Refer to Figure 6-14 below.

Example B illustrates two power objects named VCC and VDD connected to a +5 volt power supply through a power object named +5V.

Example C illustrates a power object named VEE that is connected to a -5.2 volt power supply through a power object named -5.2V.

Example D illustrates a power object named VDD connected to a power object named GND. This electrically connects the two types of grounds in the net list.

(No Figure !)

Figure 6-14. Power Supply with a Power Object

6.7.17. Handling and Isolating Grounds

The NETLIST program will treat certain parts in the library as a power object if they are defined a special way. The four types of grounds in the DEVICE library are good examples. They are Earth, Field, Power, and Signal grounds. To be treated as power objects, a device is defined with zero parts per package and having only one pin defined. The one pin is a PWR type, as illustrated below in the example of the Power ground symbol. When these conditions are met, the NETLIST program internally represents this part in the same maner as a power object.

```
'GND POWER'
(X Size  =) 2  (Y Size =) 1  (Parts per Package =) 0
T1   PWR       'GND'

(  0) ####################
(  1) ....................
(  2) ....................
(  3) ...################...
(  4) ....................
(  5) ....................
(  6) ......#########......
(  7) ....................
(  8) ....................
(  9) .........###.........
( 10) ....................
```

In the above example, notice that the pin name is called GND. When a Power ground symbol is placed in the worksheet,

NETLIST interprets this ground symbol to be connected to any other object that has a power pin named GND.

To isolate the different types of grounds, change the pin name in the library source file. For example, rename GND in a Signal ground library source to SGND. Or rename GND in the power ground library source to PGND. The NETLIST will then interpret each type of ground to be connected with any other object that has a power pin defined as SGND or PGND.

6.7.18. Connecting Power Objects to a Module Port

There are the cases when you wish to isolate power in the worksheet. One way to do this would be make all new parts with new power pin names. Not only would this be time consuming, but it would be very difficult to keep track of which parts are to be used on which schematic for any particular supply.

Isolating power without having to create new parts is done by connecting a module port to a power object. The NETLIST program will then supersede the use of the power object with a module port. Only the module port will be passed off of a sheet to isolate circuitry used on another worksheet.

If a power object is to transfer power from one worksheet to another, either in a flat file or hierarchical structure, it must be connected to an "unspecified" module port. Anu other type of module port is not accepted by ERCHECK. Three examples of connecting power objects to module ports are illustrated in Figure 6-15 below.

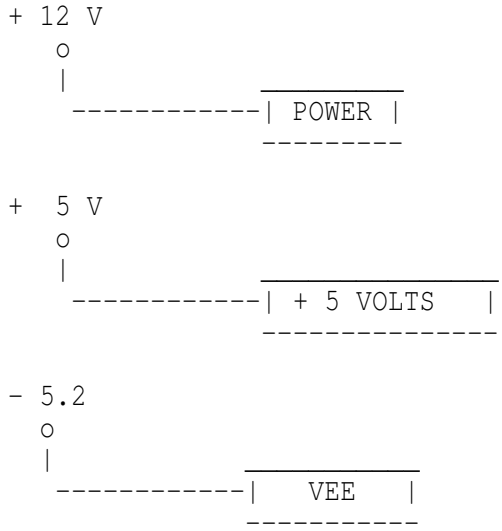


Figure 6-15. Connecting Powet Objects to Unspecified Module Ports

.pa

6.7.19. Considerations When Handling Power in a Hierarchy

Power in a hierarchy is handled in much the same manner as a flat file structure. Power objects will connect to all other objects that have the same name. If a module port is connected to a power object, then the module port will supersede the power

object in going off of the sheet. Note, that this module port is treated the same as all other module ports.

It is necessary to make a sheet net in the sheet symbol when passing power in a hierarchy. If a sheet net is not defined, the NETLIST program will not resolve the connections of the module port properly.

6.7.20. Example of Isolating Power - Battery Backup

A power object connected to a module port provides an easy way to isolate power in a design. In battery backup applications, main power can be supplied throughout the design with power objects. Backup power can be isolated from the main source by using a module port. For example, Figure 6-16 through 6-18 illustrate a battery backup application.

This design is was created as a three sheet hierarchy. The root sheet, in Figure 6-16, contains the CPU and control circuitry of the design. Two hierarchical sheet symbols are also placed in the root worksheet. One sheet symbol represents the power supply, the other represents the memory that is to be battery backed up.

Notice that a VDD power object is placed in the root worksheet and is connected to a +5V power object. Since the 80C51 and the 82C82 power pins are labeled as VDD in their library source file, the +5V and VDD power objects will connect plus 5 volts from the power supply (Figure 6-17) to the VDD pins in the NETLIST output.
}PDRAFT}.PN157

Figure 6-18 illustrates the CMOS memory that is to be battery backed up. The memory control signals are transferred from the CPU sheet through module ports AD[0..7], WE, and A[0..7]. The power supplied to the CMOS MEMORY worksheet is insolated from the +5V supply through a module port labeled:BACKUP

In the CMOS MEMORY sheet, the "BACKUP" module port is connected to a power object named VDD. The power object named VDD is global only to the worksheet that it is placed in. This means that signals will connect to all other signals that have the same name, VDD in this case. Power remains insolated from VDD on the CPU sheet through the use of the BACKUP module port.

Notice, that a GND and VSS power object is also placed in the CMOS MEMORY worksheet. This connects the VSS power return pins from the memory devices to the Power ground object.

To summarize, power may be insolated in a design by transferring it through module ports that are connected to selectively named power objects that match the power pin names in library source files.

Although, this design was created as a hierarchy, it also could

have been created as a flat file structure. In applications where you insolate power, always place the circuitry to be insolated in a separate worksheet. This keeps the insolated power specific to one worksheet.

6.7.21 Handling Physical Connectors

Module ports are not intended to be used as physical connectors in your design. If they are used, the NETLIST utility has no way to connect signals to specific pin numbers. This is because module ports do not have pin numbers. Physical connectors should always be made as a library part, since it is desired to have a reference designator and part value associated with the connector.

Easy-to-read schematics should refrain from separating individual pins in a connector and placing them all over the worksheet. This makes locating connector pins a real task, espically when multiple sheet are envolved. It is recommended to place a connector in one worksheet and use module ports or labels to connect pins to other signals in the design. Figure 6-19 illustrates this below.
.pa 0
3/9

Making large connectors with alphanumeric pin names can easily be handled by making the connector as a block symbol. To make a connector with alphanumeric pin names, make the part wirh zero "0" parts per package. Do not use the token GRIDARRAY or "1" part per package. Listed below is a library source example of an IBM 62 pin edge connectors as. This is only a partial example.

```
'CONNECTOR IBM'
REFERENCE  'J'
10    32     0
L1    IN     'B1'
L2    IN     'B2'
L3    IN     'B3'
L4    IN     'B4'
      .
      .
      .
L30   IN     'B30'
L31   IN     'B31'
R1    IN     'A1'
R2    IN     'A2'
R3    IN     'A3'
R4    IN     'A4'
      .
      .
      .
R30   IN      'R30'
R31   IN      'R31'
```

6.7.22 CONFIG.SYS File

If you get "MSDOS error #4 NO HANDLES LEFT", be sure that your computer CONFIG.SYS file has these parameters:

```
     BUFFERS = 16
     FILES = 10
.PA O+/9
```


6.8 PARTLIST

Purpose:

Creates a summation of all parts used in a group of schematic sheets. The PARTLIST program will scan either a hierarchy, flat file, one sheet file structure, or the annotation from the ANNOTATE program.

Optionally, user specific part information may be added in a text, or "include file". If this user specific information is included, then the PARTLIST program willoutput the parts found in the order that they appear in the include file. Any parts not in the include file are placed at the end of the report.

Format:

PARTLIST source [destination] [include] [/A] [/C] [/E] [/F] [/I] [/O] [/Q] [/S] [/V]

Remarks:

The source may be either the root sheet name of a hierarchical file structure, the name of a text file in a flat file structure, or the file name of a one sheet file structure.

If the source is the name of a text file in a flat file structure, the /F switch must be included. If the source is the file name of a one sheet file structure, the /O switch must be included. Ifthe source is the annotation file created by the ANNOTATE program, then the /A switch must be included on the invocation line.

The destination is any valid DOS path name and is where the output of the program is to be placed. If a destination is not specified, the output of the PARTLIST program is directed to the console.

The include is the part name of a text file which contains user information to be included in the part list. The format of this file is discussed later.
This parameter is only recognized when the /I switch is present on the command invocation line. If the include file is used, you must follow the format illustrated above when invoking the PARTLIST program. If there are two path names on the invocation line, the first path name will be used as the source and the /I switch will cause the second path to be used for the include path. If there are three path names on the invocation line, then

the third path will be used for the include path.

The /A switch causes the PARTLIST program to read the source path as an annotation file.

The /C switch causes the configuration menu to be invoked. This allows the OrCAD/SDT environment to be modified.

The /E switch causes the utility program to display the message "Type Any Key To Continue", enable the system to pause for you to remove the DRIVER/LIBRARY disk in drive B and insert the SHEET disk. This switch is used only on systems with two floppy disk drives.

The /F switch causes the PARTLIST program to read the source as a text file, for flat file structure applications.

The /I switch causes the PARTLIST program to include user supplied information in the output.

The /O switch causes the file name of the source to be read as a one sheet file structure.

The /Q switch causes the PARTLIST program to run "quietly". This means that only the invocation messages and error messages if any, are displayed. If this switch is not specified, the program will display intermediate tracking activity.

The /S switch causes the format of the output report to be single spaced rather than defaulted to double spaced.

The /V switch output a verbose format, whitch causes the header information to be included on every page.

6.8.1 Invocation Examples Using Hierarchical Structured Files

1. To create a parts list of a hierarchical schematic:

```
PARTLIST root.sch
```

Where root.sch is the path and name of the root sheet in the hierarchy.

2. To obtain a parts list a sub-sheet in a hierarchical schematic:

```
PARTLIST subsheet.sch/O
```

Where subsheet.sch is the path and name of the sub-sheet in the hierarchy, /O signifies that the subsheet.sch file name is a single sheet.

3. To direct the output of the PARTLIST program to a file:

PARTLIST root.sch whatfile

Where root.sch is the path and name of the root sheet in the hierarchy and whatfile is the path and file name to place the PARTLIST information.

4. To direct the output of the PARTLIST program to a file and include additional information: o=/9

PARTLIST root.sch whatfile include.txt /I

Where root.sch is the path and name of the root sheet in the hierarchy, whatfile is the path and file name to place the PARTLIST information, include.txt is the name of the include text file, and /I signifies that a include file is specified.

6.8.2 Invocation Examples Using Flat File Structures

1. To creae a parts list of a flat file structure containing multiple sheets:

PARTLIST flatfile.txt /F

Where flatfile.txt is a text file containing a list of schemartic file names to be ploted, /F is used to signifi that flatfile.txt is a text file.

2. To create a part list of one sheet in a flat file structure:

PARTLIST sheetname.sch /O

Where sheetname.sch is the name of the single sheet in the flat file structure, /O is used to signify that sheetname.sch is a single sheet schematic.

3. To create a parts list of a flat file structure and direct the output to a file:

PARTLIST flatfile.txt whatfile /F

Where fletfile.txt is a text file containing a list of schematic file names to be checked, whatfile is the path and file name to place the PARTLIST information. /F is used to signify that flatfile.txt is a text file.

4. To create a parts list of a flat file structure, direct the output to a file, and include additional information:

PARTLIST flatfile.txt whatfile include.txt /F /I

Where flatfile.txt is a text file containing a list of schematic file names to be checked, whatfile is the path and file name to place the PARTLIST information, include.txt is the name of the include file, /F is used to signify that flatfile.txt is a

text file, and /I signifies that an include file is specified.

6.8.3 Invocation Examples Using A One Sheet File Structure

1. To create a parts list of a single sheet schematic:

```
PARTLIST sheetname.sch /O
```

Where sheetname.sch is the name of the single sheet

schematic, /O is used to signify that sheetname.sch is a single sheet schematic.

2. To create a parts list or a single schematic and direct the output to a file:

```
PARTLIST sheetname.sch whatfile /O
```

Where sheetname.sch is a name of the single sheet schematic, whatfile is the path and file name to place the PARTLIST information, and /O is used to signify that sheetname.sch is a single sheet schematic.

3. To create a parts list of a single schematic, direct the output to a file, and include additional information:

```
PARTLIST sheetname.sch whatfile include.txt /O /I
```

Where sheetname.sch is the name of the single sheet schematic, whatfile is the path and file name to place the PARTLIST information, include.txt is the name of the include file, /O is used to signify that sheetname.sch is a single sheet schematic, and /I signifies that an include file is specified.

6.8.4 Obtaining a Parts List from Schematics Based on an Annotation File

```
PARTLIST annotation.out /A
```

Where annotation.out is the output from the ANNOTATE program, /A causes the PARTLIST program to read annotation.out as an annotation file.

6.8.5 Include File Format

The include file is a simple text file that you create, in which you place additional part information. The first line of the file contains a header line which will be added to the header placed at the top of each page.

The header line begins with a pair of single quotes with no characters or space between them. The remainder of the line

contains the header information that you want to include. For each part to include information, a separate line in the file is created which begins with the part name as it appears in your worksheet. The name must be enclosed within single quotes (such as '74LS00').

After you enter the part name, place the information on the same line that you want to be included ("Resistor 1/4 Watt 5%" for example). For both types of lines, header and part, the line will be left justified to the first non-space character of the information portion of the line. When the PARTLIST program has o=/9

---

finished seanning the sheets, it then scans the include file to include the rest of the line after any part name that matches. The following is a sample of the include file format.

```
'           DESCRIPTION                 Part Order Code
'1K'        Resistor 1/4 Watt 5%        10000111003
'4,7K'      Resistor 1/4 Watt 5%        10000114703
'22K'       Resistor 1/4 Watt 5%        10000112204
'1uf'       Capacitor Ceramic Disk      10000211006
'.1uf'      Capacitor Ceramic Disk      10000211007
'.01uf'     Capacitor Ceramic Disk      10000211008
'.001uf'    Capacitor Ceramic Disk      10000211009
'7400'      TTL Quad Two Input NAND Gate 10001040000
'74LS00'    TTL Quad Two Input NAND Gate 10002040000
.pn164
```

```
'74S00'    TTL Quad Two Input NAND Gate   10003040000
'74ALS00'  TTL Quad Two Input NAND Gate   10004040000
'74AS00'   TTL Quad Two Input NAND Gate   10005040000
'7402'     TTL Quad Two Input NAND Gate   10001040002
'74LS02'   TTL Quad Two Input NAND Gate   10002040002
'74S02'    TTL Quad Two Input NAND Gate   10003040002
'74ALS02'  TTL Quad Two Input NAND Gate   10004040002
'74AS02'   TTL Quad Two Input NAND Gate   10005040002
```

NOTE

The PARTLIST utility does not check for duplicate entries in the include file. If one exists, the program may become suspended.

The Include file is also case and trailing-character sensitive. For example: if a part is labeled: 2.5 mH in your worksheet, PARTLIST does not make a match if your Include File has the part as: 2.5 MH.

6.8.6 Sample PARTLIST Output

```
Bill Of Materials September 15, 1986  14:48:48           Page 1

Item      Quantity     Reference              Part
-----------------------------------------------------------------
 1          1           U1                     8051
 2          1           X1                     12 mHz
 3          2           C1, C2                 30 pf
 4          1           C3                     20 uF
 5          1           U3                     2732
 6          1           U4                     8282
 7          2           R2,R4                  6.8K
 8          1           R3                     470
 9          1           U5                     74LS73
10          1           U6                     74LS86
11          1           FUSE1                  2 AMP
12          1           T1                     SC140
13          1           R8                     47
.PA
PARTLIST                           Schematic Design Tools
-----------------------------------------------------------------

14        2         D2,D3              1N4004
15        2         Q2,Q3              TIP110
16        1         R9                 5 ohm
17        1         SW1                SpSt
```

6.8.7 Sample PARTLIST Output Using an Include File

```
-----------------------------------------------------------------
Motor Drive Circuitry                      Revised: June 22, 1986
786-256A-001                               Revision: 1
Bill Of Materials September 15, 1986  14:48:48         Page 1
Item Quantity Reference Part DESCRIPTION Part Order Code
-----------------------------------------------------------------
 1   1   U1     8051      Intel Controller        10002048505
 2   1   X1     12 mHz    Crystal                 10000820006
 3   2   C1,C2  30 pf     Capacitor Mica          10000483736
 4   1   C3     20 uF     Capacitor Tantalum      10000486353
 5   1   U3     2732      32K EPROM               10002734645
 6   1   U4     8282      Latch                   10008475663
 7   2   R2,R4  6.8K      Resistor  1/4 Watt 5%   10038437622
 8   1   R3     470       Resistor  1/4 Watt 5%   10038437862
 9   1   U5     74LS73    J-K Flip Flop           10008756353
10   1   U6     74LS86    2 input EX-OR Gate      10008756349
11   1   FUSE1  2 AMP     Slow Blow Fuse          15000063731
12   1   T1     SC140     2 AMP Triac             10040000295
13   1   R8     47        Resistor 1/4 Watt 5%    10038437023
14   2   D2,D3  1N4004    Diode                   10092735660
15   2   Q2,Q3  TIP110    Power NPN               12000838388
16   1   R9     5 ohm     Resistor 1/4 Watt 5%    10038430005
17   1   SW1    SpSt      B&K 100 MA Switch       10842100954
.pa
PLOTALL                            Schematic Design Tools
-----------------------------------------------------------------
```

6.9 PLOTALL

Purpose:

Plots a group of schematic sheets which may be a hierarchy, flat file, one sheet file structure, on an annotation file. As an option, grid reference may be specified.

Format: PLOTALL source [destination] [/A] [/C] [/E] [/F] [/G] [/O] [/Q]

Remarks:

The source may be either the root sheet name of a hierarchical file structure, the name of a text file in a flat file structure, or the file name of a one sheet file structure.

If the source is the name of a text file in a flat file structure, the /F switch must be included. If the source is the file name of a one sheet file structure, the /O switch must be included. If the source is the annotation file created by the ANNOTATE program, then the /A switch must be included on the invocation line.

The destination is any valid DOS path name and is where the output of the program is to be placed. If a destination is not specified, the output of the PLOTALL program is directed to the serial channel specified in the plotter configuration (refer to Section 2 for configuration information).

The /A switch causes the PLOTALL program to read the source path as an annotation file.

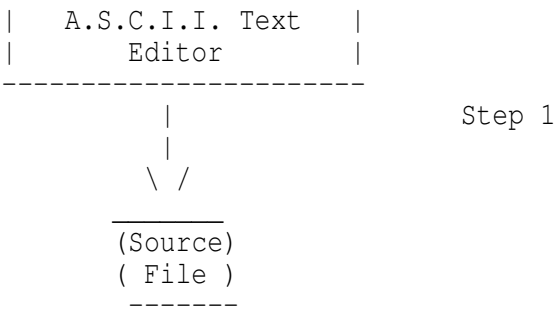
The /C switch causes the configuration menu to be invoked. This allows the OrCAD/SDT environment to be modified.

The /E switch causes the utility program to display the message "Type Any Key To Continue", enabling the system to pause for you to remove the DRIVER/LIBRARY disk in drive B and insert the
.pa

SHEET disk. This switch is used only on system with two floppy disk drives.

The /F switch causes the PLOTALL program to read the source as a text file, for flat file structure applications.

The /G switch causes GRID REFERENCES to be includtic Design Tools Libraries



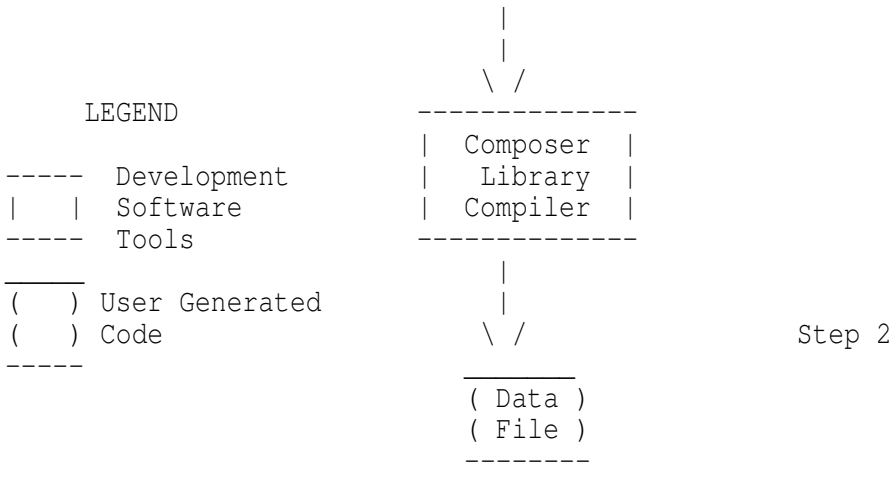


Figure 7-1. The Library Development Process

OrCAD systems supplies a number of part libraries for use with DRAFT. Part of configuring DRAFT means choosing what libraries it will have acess to. The libraries you choose at configuration time are loaded into RAM when you invoke DRAFT. This eliminates disk searching and provides for quick part retrieval.

You can choose from OrCAD supplied librariesas well as your own custom libraries to load into DRAFT. The custom libraries that you create will behave just like the OrCAD supplied libraries.

OrCAD-upplied libraries are shipped as library data files. This is for your convenience (data file are ready to use) and also to save disk space. A library source file takes up much more disk space than its corresponding library data file. A source file can be four or five times as large as its data file.

By applying DECOMP to an OrCAD-supplied library, you can construct a source version of that library. This feature is useful if the library you want to create has parts similar, but not identical, to those in the OrCAD-supplied library. With a text editor, you can extract the similat parts and edit them. The supplied libraries also serve as an extensive list of examples. If, while creating your own library, you get stuck, you can always look inside a source version of a supplied "library to see how OrCAD did it."

Creating custom libraries is preferable to modifying OrCAD-supplied libraries. If you modify an existing library, you run the risk of future OrCAD updates negating your changes. If is not good practice to run DECOMP on an OrCAD-supplied library, edit the resulting source file, and then run COMPOSER and replace the original library.

You may have parts with the same name in different libraries. If you do and those libraries are selected. DRAFT searches through the libraries in the order you specify at configuration time.

7.2. The COMPOSER and DECOMP Utilities

The COMPOSER utility is called COMPOSER.EXE and the DECOMP utility is called DECOMP.EXE. Both files are found on the LIBRARY disk. To execute either utility, you must be at the DOS command level in the directory that contains the utility you want to execute.

7.2.1. Invoking COMPOSER

COMPOSER source library <ENTER>

where: source is the name of the ASCII text file that describes your custom parts using OrCAD's Synbol Description Language.
The .SRC extension is a convention, not a requirement.
library is the name of the resulting library file. If you give the name of an existing file, COMPOSER asks if you want to overwrite the existing file. You cannot append to an existing file. <ENTER> represents the ENTER key on your computer. DOS commands are executed when you type the ENTER.

Both source and library may be complete pathnames -- that is, if either is in a directory other than your current directory, you must specify the complete path. Here are two examples.

COMPOSER custom.src custom.lib <ENTER>

The files COMPOSER.EXE and custom.src are in the same directory, and this directory is your working directory. This example creates the file custom.lib in your working directory.

COMPOSER \orcad\library\custom.src \orcad\library\custom.lib <CR>

The file COMPOSER.EXE is in your working directory, which is not necessarily \orcad\library. This example creates the file custom.lib in the directory \orcad\library.

7.2.2. Invoking DECOMP

DECOMP library source <ENTER>

where: library is the name of an existing library file. The .LIB extension is a convention, not a requirement, source is the name of the resulting ASCII text file that describes the parts in the specified library. Comments are included in the file for clarity. If you give the name of an existing file, DECOMP asks if you want to overwrite the existing file. <ENTER> represents the ENTER key on your computer. DOS commands are executed when you type the ENTER.

As with COMPOSER, both source and library, may be complete pathnames -- that is, if either is a directory other than your current directory, you must specify the complete path.

Here are two examples.

DECOMP custom.lib custom.src <ENTER>

The files DECOMP.EXE and custom.src are in the same directory, and this directory is your working directory. This example creates the file custom.src in your working directory.

DECOMP \orcad\library\custom.lib \orcad\library\custom.src <CR>

The file DECOMP.EXE is in your working directory, which is not necessarily \orcad\library. This example creates the file custom.src in the directory \orcad\library.

Note that DECOMP does not return your source; it makes its own. For example, the comments in your original source are not reproduced. DECOMP adds its own comments. Also, DECOMP may rearrange the order of the part definitions. DECOMP lists parts in numeric order followed by parts in alphabetical order. For example, assume that you define two pats, one called resitor and one called 7400. You place resistor in your source file before 7400, run COMPOSER, then DECOMP to produce a new source file. Unlike your original source file, the new source file has 7400 listed before resistor.

7.3. Creating a Source File

A source file consists of a prefix definition followed by a series of part definitions. You can have only one prefix definition per library, and it occurs at the beginning of the library. There are two types of part definitions: block sTools

---

This assumes that you are using serial channel 1 (COM1) and have your plotter set for 2400 baud. For more information on the MODE command, refer to your DOS users guide for Asynchronous Communications.

After the serial channel has been configured, send the plot file to the plotter using the DOS COPY command as follows:

COPY whatfile COM1: <ENTER>

Where whatfile is the name of the plot file.

If the plotter works, this indicates the problem may be in the plotter cable (incorrectly wired), or the hardware handshaking is incorrectly set (check PLOTALL configuration).

If the plotter does not work, this indicates that there is a hardware problem. Check the following: serial card, incorrect serial channel configuration, plotter hardware, or a cable problem.

6. If yours is an IOline plotter, be sure you have PROM version 114 or greater.

6.9.9 Output Scaling

OrCAD does not control the scale of the plotter output. Output

scaling is controlled entirely by your plotter. Plotter scaling is typically controlled by the size of the paper used, P1 and P2 point settings (HP plotters), rotation settings, and the default settings on the plotter.

If you change the size of the worksheets you are plotting, (plotting a C size, then a B size worksheet for example) always RESET the plotter when changing worksheet size. If you have further scaling questions, refer to your specific plotter manual.
.PA
.PN171

6.9.10 Plotter Hints

When making a plot, use the proper pens and paper designed for the plotter. Plotter paper has a "memory" to it. If it hangs on the plotter bed for a period of time, it will stretch. This effects the registration of the plot. Plotter paper is also temperature sensitive. Be sure that the paper is at room temperature before plotting. The longer the drawing takes to plot, the more care must be exercised with the paper.

The configuration of the plotter includes the ability to change the velocity of the pens. When the pen can not draw at the speed the plotter is capable of moving, reduce the velocity. You will need to consult your plotter manual for the range to set the velocity. The velocity can be set only in whole number values.

When you make a plot with different pens, the plotter has a registration inaccuracy that must be considered. If you wish to have the highest quality plot, always use only one pen.

When you make a plot on a paper size that does not match the worksheet size, the plotter drivers will scale the drawing to fit the paper selected. We do not recommend making a plot more than one size off of the worksheet size since the width of the pen is fixed. For example, you can plot a C size worksheet on B, C, or D and it will work fine. If you plot an E worksheet on A paper, you will not be able to read the writing. If you plot an A worksheet on size paper, the bit mapped devices will be "grainy".

When you are directed by the program to chance paper or pens, always wait until the plotter has finished the present plotting activity. Before sending a plot directly to the plotter, be sure that the plotter is online, the pen(s) are properly set up, and the paper size is correct. When you have a pen that must be manually changed, the PLOTALL program will pause and inform you of the object to be plotter with the new pen.

6.9.11 HP Plotters

The HP plotter family has a facility to det the corner points of the plot and automatically scale the plot to be within these

points. These points are called P1 and P2.

When you change the paper sie, or at power up, these points are set to default values which depend on the size of the paper and the plotter's margin requirements. You may reset P1 and P2 to the outer most boundary of the paper and the plot will be slightly larger.

If you plot on paper that is not the same size as the worksheet, you will have to manually adjust the location of P1 and P2 to correct for the aspect ratio.

6.9.12 HI Plotters

The HI plotter family does not automatically scale the drawing. If you direct the output of the PLOTALL program to a file you will be aspect ratio is correct for both direct-to-plotter and re-direct-to-file plots.

The HI 40 Series defaults to 2400 baud, and the 50 Series defaults to 9600.

Always check to make sure that the plotter baud rate, and data bit settings correspond to the plotter configuration (refer to "Setting Up The PLotter Configuration" above).

If data and parity do not match, the message: "<<<error>>> Unable to read back from plotter" appears on the screen.

Be sure that the plotter is on-line before beginning a plot. The HI plotters do not have a means to set the velocity to the power-up default. If you change any of the velocity setting of the pens in the configuration, you will need to set them all. The velocity ranges can be found in the plotter operation manual for you specific plotter.

6.9.13 Suppressing the Title Block and Border

To suppress the title block and border of the worksheet, invoke the Configuration Menu by entering "DRAFT/C" from the DOS command line.

Press <C> <T> to obtain the "Color Table/Plotter Pen Table". At the "Command ->" prompt, press the <P> <M> keys. Then, press <9> followed by the <ENTER> key. The plotter pen is now IGNORED for drawing the worksheet title block and border.
.PA

6.10 PRINTALL

Purpose:

Prints a group of schematic sheets which may be a hierarchy, flat file, one sheet file structure, or an annotation file. As options, grid references, scaled output, and wide paper can be

specified.

Format:

PRINTALL source [destination] [/A] [/C] [/E] [/F] [/G] [/O] [/Q] [/S] [/W]

Remarks:

The source may be either the root sheet name of a hierarchical file structure, the name of a text file in a flat file structure, or the file name of a one sheet file structure.

If the source is the name of a text file in a flat file structure, the /F switch must be included. If the source is the file name of a one sheet file structure, the /O switch must be included. If the source is the annotation file created by the ANNOTATE program, then the /A switch must be included on the invocation line.

The destination is any valid DOS path name and is where the output of the program is to be placed. If a destination is not specified, the output of the PRINTALL program is directed to the printer PRN.

The /A switch causes the PRINTALL program to read the source path as an annotation file.

The /C switch causes the configuration menu to be invoked. This allows the OrCAD/SDT environment to be modified.

The /E switch causes the utility program to display the message "Type Any Key To Continue", enabling the system to pause for you to remove the DRIVER/LIBRARY disk in drive B and insert the SHEET disk.
.pa

This switch is used only on systems with two floppy disk drives.

The /F switch causes the PRINTALL program to read the source as a text file, for flat file structure applications.

The /G switch causes GRID REFERENCES to be included in the sheet printout.

The /O switch causes the file name of the source to be read as a one sheet file structure.

The /Q switch causes the PRINTALL program to run "quietly". This means that only the invocation messages and error messages if any, are displayed. If this switch is not specified, the program will display intermediate tracking activity.

The /S switch causes the PRINTALL program to generate scaled output. If this switch is not specified, then the printer will print the worksheet in compressed mode.

The /W switch causes the printing to formatted for wide paper.

With this switch, the printing will be setup for 13 "wide paper based on the parameters of the printer driver. If the /W switch is not present, then the printing will be formatted for 8" paper.

NOTE

Epson MX printers cannot print in compressed mode. Therefore, printer output will always be in scale mode even if you attempt to print compressed.

6.10.1 Invocation Examples Using Hierarchical Structured Files

1. To print sheet in a hierarhical schematic:

PRINTALL root.sch/S

Where root.sch is the path and name of the root sheet in the hierarchy.
.pa

2. To print a sub-sheet in a hierarchical schematic:

PRINTALL subsheet.sch /O /S

Where subsheet.sch is the path and name of the sub-sheet in the hierarchy, /Q signifies that the subsheet.sch file name is a single sheet.

6.10.2 Invocation Examples Using Flat File Structures

1. To print a flat file structure containing multiple sheets:

PRINTALL flatfile.txt /F

Where flatfile.txt is a text file containing a list of schematic file names to be printed, /F is used to signify that flatfile.txt is a text file.

2. To print one sheet in a flat file structure:

PRINTALL sheetname.sch /O

Where sheetname.sch is the name of the single sheet in the flat file structure, /O is used to signify that sheetname.sch is a single sheet schematic.

6.10.3 Invocation Examples Using A One Sheet File Structure

1. To print a single sheet schematic:

PRINTALL sheetname.sch /O /S
.pa

Where sheetname.sch is the name of the single sheet schematic, /O is used to signify that sheetname.sch is a single sheet schematic.

2. To direct the output of the PRINTALL program to a file:

```
PRINTALL sheetname.sch whatfile /O
```

Where sheetname.sch is the name of the single sheet schematic, whatfile is the path and file name to place the PRINTALL information, and /O is used to signify that sheetname.sch is a single sheet schematic.

NOTE

Since whatfile is a binary print file, it will consume an extensive amount of disk space.

The file whatfile may be sent to a printer using the DOS COPY Command. For example, enter the following at the DOS prompt:

```
COPY whatfile  prn: /b
```

For additional information on the COPY Command, refer to your DOS User Manual.

6.10.4 Printing Schematics Based on the annotation file information:

```
PRINTALL annotation.out /A /S
```

Where annotation.out is the output from the ANNOTATE program, /A causes the PRINTALL program to read annotation.out as an annotation file.
.pa

6.11 TREELIST

Purpose:

Scans a hierarchical file structure of schematics to display the sheet names, sheet path names, and optionally the date of last modification. This utility is useful for organizing and keeping track of the hierarchical worksheets.

Format:

```
TREELIST source [destination] [/C] [/D] [/E] [/Q]
```

Remarks:

The source is the name of the root sheet of a hierarchical organization of schematics. Only a sheet name wil be accepted.

The destination is any valid DOS path name and is where the output of the program is to be placed. If a path is not specified, the output will be directed to the console.

Libraries are not required to be loaded for operation of this utility program. Therefore, the /E switch is not required.

The /C switch causes the configuration menu to be invoked. This allows the OrCAD/SDT environment to be modified.

The /D switch causes he program to include the date of last modification in the output for each sheet scanned in the hierarchy. If a sheet symbol is encountered but the physical sheet has not been created, then the program will display, in place of the date, <empty worksheet>.

The /E switch causes the utility program to display the message "Type Any Key To Continue", enabling the system to pause for you to remove the DRIVER/LIBRARY disk in drive b and insert the SHEET disk.
.pa

This switch is used only on system with two floppy disk drives.

The /Q switch causes the TREELIST program to run "quietly". This means that only the invocation messages and error messages if any, are displayed. If this switch is not specified, the program will display intermediate tracking activity.

6.11.1 Invocation Examples Using Hierarchical Structured Files

1. To create a hierarchical schematic tree list:

```
TREELIST root.sch
```

Where root.sch is the path and name of the root sheet in the hierarchy.

2. To direct the output of the TREELIST program to a file:

```
PRINTALL root.sch whatfile
```

Where root.sch is the path and name of the root sheet in the hierarchy, and what file is the path and file name to place the TREELIST information.
.pa

6.11.2 Sample Output from the TREELIST Utility

```
Scanning  "SHEET\286sys.sch"
Scanning  "SHEET\286proc.sch"
Scanning  "SHEET\286io.sch"
Scanning  "SHEET\286hd.sch"
Scanning  "SHEET\286fd.sch"
Scanning  "SHEET\286lan.sch"
Scanning  "SHEET\286gd.sch"
Scanning  "SHEET\286mem.sch"
```

```
<<<Root File>>>
[286sys.sch]   February 23, 1986
     Procesor and Control
     [286proc.sch]  August 28, 1986
     Peripheral Interface
     [286io.sch]    January 20,1986
     Hard Disk Interface
     [286hd.sch]    October 7,1986
     Floppy Disk Interface
     [286fd.sch]    December 6, 1986
     LAN Interface
     [286lan.sch]   <empty worksheet>
     Graphics Display Subsystem
     [286gd.sch]    May 31, 1986
Memory Array
[286mem.sch]   June 16, 1986
```

DISCUSSION

The first eight lines illustrate the TREELIST program scanning each sheet in the hierarchy retrieving pertinent information.

Text enclosed within brackets [] represents the file name of the hierarchical sheets, [286sys.sch] for example. The date represents the date of the last sheet modification.

Text placed below the worksheet file name and date, on the same indentation level, represent the names of sheet symbols that are placed in worksheet. "Processor and Control" and "Peripheral Interface" for example, are sheet symbols placed in the [286sys.sch] worksheet. [286proc.sch] is the file name that represents the "Processor and Control" sheet symbol and [286io.sch] is the file name that represents the "Peripheral Interface" sheet symbol.
.pn180

7. L I B R A R I E S

This section explain how to create your own "custom" part libraries for use with OrCAD/SDT. Section 7 describes the COMPOSER and DECOMP utilities. Then, it describes the steps you would go through to create a custom library and provides numerous examples. The section then presents a formal description of OrCAD's Symbol Description Language and concludes with examples of complete library source file.

7.1. An Overview of the Library Development Process

To create a custom library, you need a text editor and the COMPOSER utility. COMPOSER takes a library source file, which you created with the text editor, and produces a library data file, readable by DRAFT. You may also find the DECOMP utility useful. DECOMP takes a library data file and produces a library source file. You can think of DECOMP as the inverse of COMPOSER. Creating a custom library consists of the following three steps, as illustrated in Figure 7-1.

1. Create a library, text, or source file. The convention is to give the file a .SRC extension. The source file is an ASCII text file that contains instructions in OrCAD's Symbols Description Language.

   You can use any text editor. The only requirement is that it produce an ASCII file without any hidden formatting characters. For example, Wordstar, in the non-document mode produces such an ASCII file.

2. Compose the source file using the COMPOSER utility. This is similat to a compilation: it produces another file, a data file readable by DRAFT. The convention is to give this data file a .LIB extension.

3. Reconfigure DRAFT to add the new library to the list of library files.

.PA

.he Schematic Design Tools Libraries

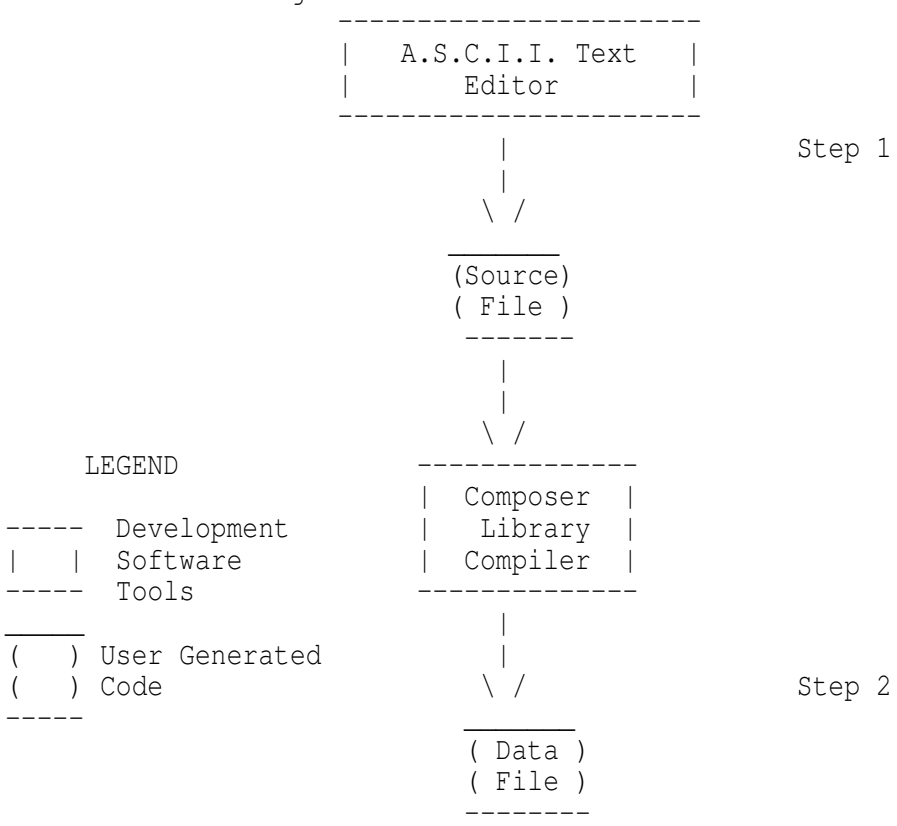


Figure 7-1. The Library Development Process

OrCAD systems supplies a number of part libraries for use with DRAFT. Part of configuring DRAFT means choosing what libraries it will have acess to. The libraries you choose at configuration time are loaded into RAM when you invoke DRAFT. This eliminates disk searching and provides for quick part retrieval.

You can choose from OrCAD supplied librariesas well as your own custom libraries to load into DRAFT. The custom libraries that you create will behave just like the OrCAD supplied libraries.

OrCAD-upplied libraries are shipped as library data files. This is for your convenience (data file are ready to use) and also to save disk space. A library source file takes up much more disk space than its corresponding library data file. A source file can be four or five times as large as its data file.

By applying DECOMP to an OrCAD-supplied library, you can construct a source version of that library. This feature is useful if the library you want to create has parts similar, but not identical, to those in the OrCAD-supplied library. With a text editor, you can extract the similat parts and edit them. The supplied libraries also serve as an extensive list of examples. If, while creating your own library, you get stuck, you can always look inside a source version of a supplied "library to see how OrCAD did it."

Creating custom libraries is preferable to modifying OrCAD-supplied libraries. If you modify an existing library, you run the risk of future OrCAD updates negating your changes. If is not good practice to run DECOMP on an OrCAD-supplied library, edit the resulting source file, and then run COMPOSER and replace the original library.

You may have parts with the same name in different libraries. If you do and those libraries are selected. DRAFT searches through the libraries in the order you specify at configuration time.

7.2. The COMPOSER and DECOMP Utilities

The COMPOSER utility is called COMPOSER.EXE and the DECOMP utility is called DECOMP.EXE. Both files are found on the LIBRARY disk. To execute either utility, you must be at the DOS command level in the directory that contains the utility you want to execute.

7.2.1. Invoking COMPOSER

```
COMPOSER source library <ENTER>
```

where: source is the name of the ASCII text file that describes your custom parts using OrCAD's Synbol Description Language.
The .SRC extension is a convention, not a requirement.
library is the name of the resulting library file. If you give the name of an existing file, COMPOSER asks if you want to overwrite the existing file. You cannot append to an existing file. <ENTER> represents the ENTER key on your computer. DOS commands are executed when you type the ENTER.

Both source and library may be complete pathnames -- that is, if either is in a directory other than your current directory, you must specify the complete path. Here are two examples.

```
COMPOSER custom.src custom.lib <ENTER>
```

The files COMPOSER.EXE and custom.src are in the same

directory, and this directory is your working directory. This example creates the file custom.lib in your working directory.

COMPOSER \orcad\library\custom.src \orcad\library\custom.lib <CR>

The file COMPOSER.EXE is in your working directory, which is not necessarily \orcad\library. This example creates the file custom.lib in the directory \orcad\library.

7.2.2. Invoking DECOMP

DECOMP library source <ENTER>

where: library is the name of an existing library file. The .LIB extension is a convention, not a requirement, source is the name of the resulting ASCII text file that describes the parts in the specified library. Comments are included in the file for clarity. If you give the name of an existing file, DECOMP asks if you want to overwrite the existing file. <ENTER> represents the ENTER key on your computer. DOS commands are executed when you type the ENTER.

As with COMPOSER, both source and library, may be complete pathnames -- that is, if either is a directory other than your current directory, you must specify the complete path.

Here are two examples.

DECOMP custom.lib custom.src <ENTER>

The files DECOMP.EXE and custom.src are in the same directory, and this directory is your working directory. This example creates the file custom.src in your working directory.

DECOMP \orcad\library\custom.lib \orcad\library\custom.src <CR>

The file DECOMP.EXE is in your working directory, which is not necessarily \orcad\library. This example creates the file custom.src in the directory \orcad\library.

Note that DECOMP does not return your source; it makes its own. For example, the comments in your original source are not reproduced. DECOMP adds its own comments. Also, DECOMP may rearrange the order of the part definitions. DECOMP lists parts in numeric order followed by parts in alphabetical order. For example, assume that you define two pats, one called resitor and one called 7400. You place resistor in your source file before 7400, run COMPOSER, then DECOMP to produce a new source file. Unlike your original source file, the new source file has 7400 listed before resistor.

7.3. Creating a Source File

A source file consists of a prefix definition followed by a series of part definitions. You can have only one prefix definition per library, and it occurs at the beginning of the library. There are two types of part definitions: block symbol definitions and bitmap symbol definitions. Comments are delimited

with braces ({}).

Block symbol definitions represents parts that are either square or rectangular. These parts are typically memory chips, microprocessors, peripheral controllers, and many TTL and CMOS devices. Bitmap symbol definitions represents parts that are complicated to define graphically. Instead, you draw them on a bitmap. They include such parts as resistors, diodes, transistors, NOSFETs, relays, and many others.

Note that lines in a source file end with a <RETURN>. The source examples in this section do not show the <RETURN>.

7.3.1. The prefix Definition

The prefix definition is delimited by the keywords, PREFIX and END. The initial delimiter is the keyword PREFIX all alone on a line. Subsequent lines contain the definition itself. The terminating delimiter is the keyword END all alone on a line.

All source file must begin with a prefix definition. If you decide your custom library doesn't need a prefix definition, you must still supply a null prefix. A null prefix consists only of the delimiting keywords. Here is how a null prefix definition looks.

```
PREFIX
END
```

On the other hand, you may find a prefix definition very useful. OrCAD specifically designed the prefix definition to handle the various TTL logic families. For example, the 74LS00, the 74S00, and the 74ALS00 have different prefixes (74LS, 74S, and 74ALS), but the same suffix (00). When you use a prefix definition, you reduce the memory required to store multiple families of parts that have different prefixes, but the same suffix.

Here is an example of a prefix definition. The example comes from OrCAD System's TTL source library, TTL.LIB.

```
PREFIX
'74LS'    = 'LS'
'74S'     = 'S'
'74ALS'   = 'ALS'
'74AS'    = 'AS'
'74HCT'   = 'HCT'
'74HC'    = 'HC'
'74ACT'   = 'ACT'
'74AC'    = 'AC'
'74F'     = 'F'
'74'
END
```

DRAFT uses the prefix definiton when you obtain a part with the Get command, instead of entering the entire name of the part, you can enter just the suffix. DRAFT displays a pop-up menu that lists all the valid part names constructed by appending the suffinx you provided with the prefixes in the prefix definition. For example, if TTL.LIB is one of your libraries and you enter

the suffix 04, the pop-up menu lists the following parts.

74LS04
74S04
74ALS04
74AS04
74HCT04
74HC04
74F04
7404

A prefix definition is constructed as follows. First, enter the PREFIX keyword followed by a <RETURN>. Then, begin the first prefix string by entering a single quote ('). Type the prefix string. It consists of a string of printable ASCII characters no more than seven characters long. DRAFT does not distinguish between uper and lower case. Close a prefix string with another single quote.

Then, enter a <SPACE> followed by an equal sign (=) followed by another <SPACE>. To improve readability, you can delimit the equal sign with any number of <SPACE>s or <TAB>s. Now enter the shorthand string. This is the path of the prefix string that varies. The shorthand string also consists of no more than seven printable ASCII characters. Then, enter a <RETURN> and type the next line. You can define a maximum of sixteen prefix strings.

The shorthand string enables you to bypass the pop-up prefix menu and still enter an abbreviated part name. For example, you can obtain the part 74HC04, by supplying the GET command with the abreviated name HC04. This is possible because HC is a shorthand string for 74HC.

7.3.2. The Part Definition

The part definition defines the part's name, its size (in unit lengths on the screen and in tenths of an inch on the printed worksheet), the number of parts per package and the pin functions (input, output, open, collector, etc.). There are two types of part definitions: block symbol definitions, and bitmap symbol definitions.

You do not have to group your block definitions and bitmap definitions together. For example, your source file may contain a block definition, followed by a bitmap definition, followed by another block definition.

Block and bitmap definition follow much the same syntax. A bitmap definition looks like a block definition followed by a bitmap. When COMPOSER sees a bitmap, it uses that bitmap to represent the part, rather than defaulting to a square or rectangle.

A symbol definition has the following fields.

* One or more part name strings. A name is a printable ASCII string enclosed in single quotes. If you have more than one part name string, delimit them with a <SPACE> or put them on separate lines. When obtaining a part, you can use any of your supplied names.

* An optional reference designator. The ANNOTATE utility automatically updates reference designators.

* The symbol size. Each unit represents a unit length on the screen and 0.1 inch on the printed worksheet. You give the X size first and then the Y size. On the same line you list the number of parts per package. If the part is a pin grid array, specify the keyword GRIDARRAY instead of the number of parts per package.

* The pin definition. Each pin is defined on a separate line. A pin definition consist of the following fields:

    * The pin position;

    * The pin number of GRIDARRAY pin name;

    * The optional DOT keyword (which places the inversion bubble at the pin position);

    * The optional CLK keyword (which places the clock symbol at the pin position);

    * The optional keyword SHORT (which places 0.1-inch leads at the pin position instead of the standard 0.3-inch leads). SHORT cannot be used with DOT or CLK;

    * The pin function (IN, OUT, I/O, OC, PWR, PAS, HIZ);

    * The pin name string.

* An optional bitmap. Use this if the symbol you want is not a square or rectangle.

* An optional convertion. This only has meaning is you've defined a bitmap. The most common use for converted bitmaps is to specify the DeMorgan equivalent of the defined part.

7.3.3. Block Symbol Definition

Illustrated below is an example of a block symbol definition. The example does not represent areal part, although is similar to a JK flipflop. Figure 7-2 shows the symbol produced by this blok definition.

.pa

```
'74EXAMP'
REFERENCE      'LATCH'
6    10   2
L1   3    11   SHORT     IN   'J'
L5   1    13   DOT CLK   IN   'CLK'
L9   2    12   SHORT     IN   'K'
B3   15   14   DOT IN          'CL'
L3   4    10   DOT IN          'P'
R1   6    7    OUT             'Q'
R9   5    9    OUT             'Q\'
T0   16   16   PWR             'VCC'
B0   8    8    PWR             'GND'
```

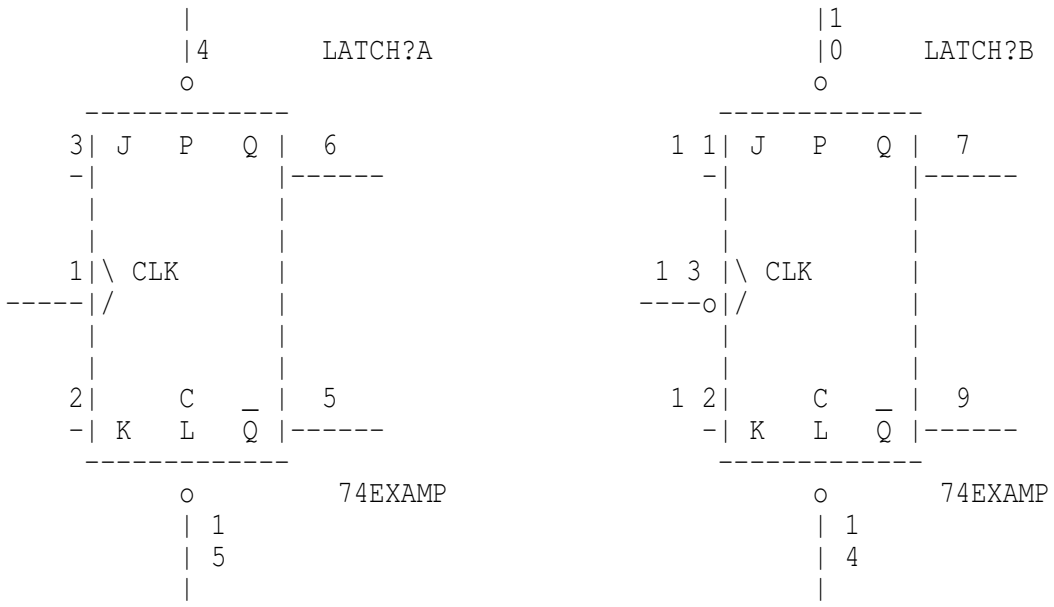


Figure 7-2: The Block Symbol for 74EXAMP

Part name string. The example has only one part name string, '74EXAMP'.

Reference designator. After the part name string comes the optional reference designator.

What appears when you call up a part is determined as follows:

1. If the device has 0 parts per package and you do not specify a REFERENCE key word, none appears. Nor does the part name appear.

2. If the device has 0 parts per package and you specify a REFERENCE key word, it appears. It consists of the string you specified followed by a question mark. ANNOTATE replaces the ? with a sequential number. The part name also appears.

3. If the device has on or more parts per package, and you do not specify a REFERENCE key word, a default reference designator (U?A) appears. ANNOTATE replaces the ? with a sequential number that identifies the occurrence of the device and replaces the A with a letter that cycles through the part device. The part name also appears.

4. If the device has one or more parts per package, and you specify a REFERENCE key word, it appears. It consists of the string you specified followed by ?A. The part name also appears.

XY size and parts/package. The next line has three numbers 6 10 2. The first two represent the size. The size of the part is 6X by 10Y, where each unit represents one screen unit or 0.1 inch on the printed worksheet. The 2 indicates that there are two parts per package. If the part were a pin-grid array, you would

supply the keyword GRIDARRAY in place of the number of parts per package.

Pin definition. The rest of the example consists of the pin definitions. Consider the second pin position. The first field L5 locates the pin on the left side of the part in the fifth position counting nnn the top down. The Y dimension specified 11 possible positions, 0 through 10. The first possible is L0 and the last is L10. The specified pin position (L5) is 0.5 inches from the top of the part, when seen on the printed worksheet.

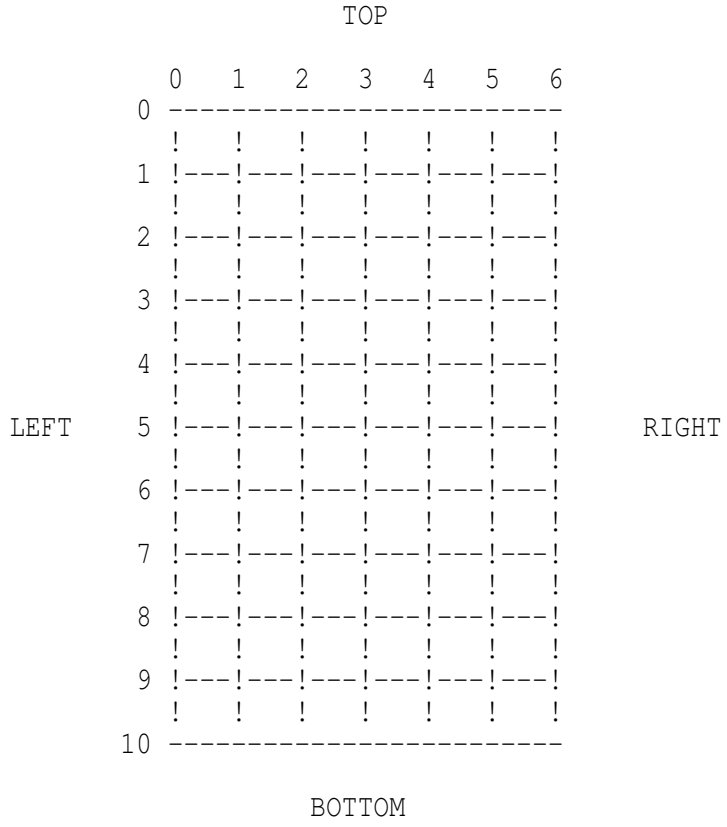


Figure 7-4. The Grid of a &X by 10Y Block Symbol

The next two pin position fields identify the pin numbers, 1 for the first part of the package and 13 for the second part of the package. The pin at L5 specifies DOT to obtain the inversion bubble and CLK to get the clock symbol. In this case, DOT and CLK are modifiers of the pin function, IN.

R9 puts a pin on the right side in the ninth position, and B3 puts a pin on the bottom in the third position counting right. The two power supply connections are at the top and bottom in the zero position.

Figure 7-4 shows a grid that represents the possible pin position for the 74EXAMP library part.

Note the SHORT keyword in the first pin position. Pins with the SHORT keyword have 0.1-inch leads rather than the default 0.3 inches. The SHORT keyword, however, cannot be used with DOT or CLK keywords. Finally, the 'CLK' gives the pin a name Power pins. One possible pin function is PWR for power. Note that power pins do not appear on the screen. The NETLIST utility, however, does

categorize all power pins that are connected to library parts.

If you want to make power pin visible, change the pin function from PWR to IN or PAS. If you do this, you may notice that the power pin overlaps other pin names on the symbol. To avoid this overlap, you must reposition the power pin in the corresponding library source file.

Parts per Package. If the device has more than one part per package, you can selectively display the pins. For example, assume you wanted to display the power pin VCC and GND, but only on the second part of the device, not on the first. You could do that by coding thr last two lines of thr block symbol as follows.

```
T0   0    16   PAS  'VCC'
B0   0    8    PAS  'GND'
```

When you place this symbol on the screen, the power pins do not display because the first column of pin locations contains a 0. When you place another symbol on the screen, it looks identical to the first. Both are called LATCH?A, and neither shows the power pins. However, if you exit DRAFT and run the ANNOTATE utility with the /M option (this causes the annotation information to be merged into the sheet directly) and then look at the sheet again with DRAFT, you'll see the two parts labeled LATCH1A and LATCH1B. The power pins appear only on the second part of the device, LATCH1B.

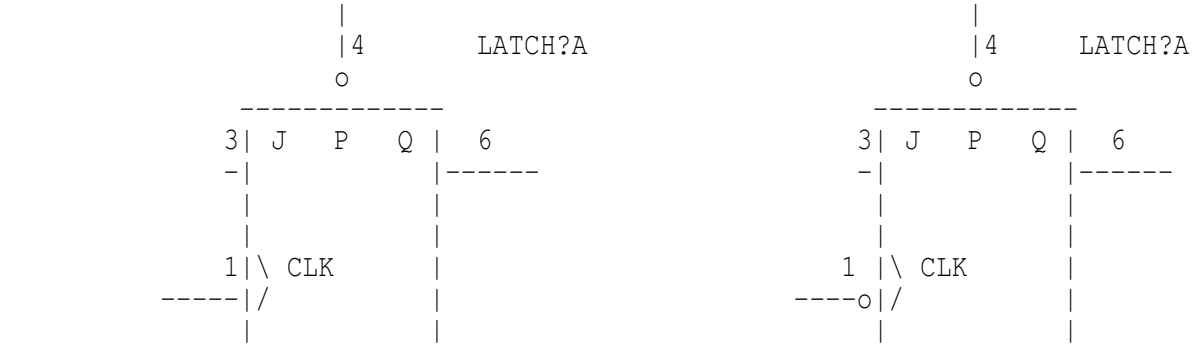
This tehnique also works for non-power pins. By specifying a pin number of 0, you can cause a pin not to appear for the part of a package.

However, if your device has one part per package, specifying a pin number of 0 does not prevent the pin from appearing. The pin appears with a pin number of 0. If your device has 0 parts per package, you cannot specify pin numbers, and consequently, none appear. Figures 7-5A and 7-5B illustrate how the number of parts per package, the pin number, and the ANNOTATE utility affect the screen symbol.

The device 74ONE is identical to 74EXAMP, except that it has one part per package. Note that the ANNOTATE utility affects both the pinout and reference designator for 74EXAMP, but only the reference designator for 74ONE. Also note that in Figure 7-5A and 7-5B, the location of the power pins were moved from T0 and B0 to R3 and R7, so that they would not overlap existing pins.

.pn191

Before ANNOTATE



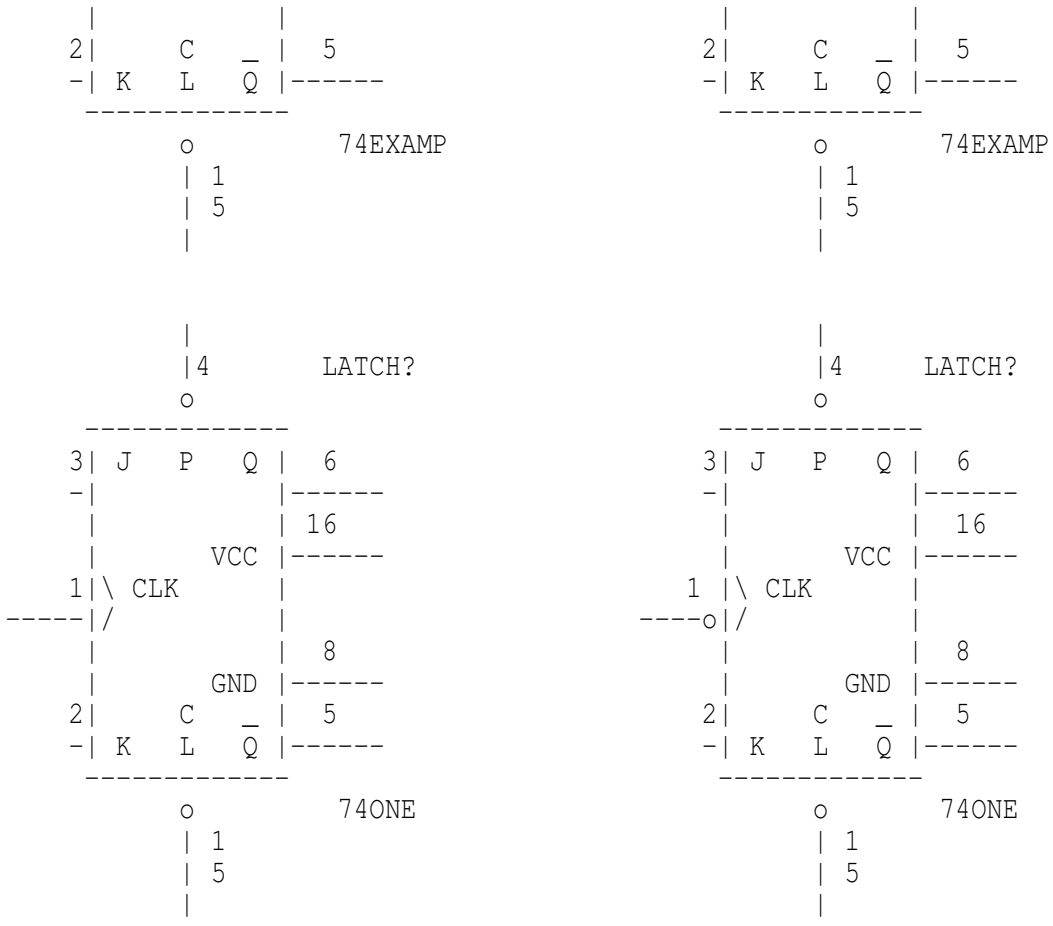
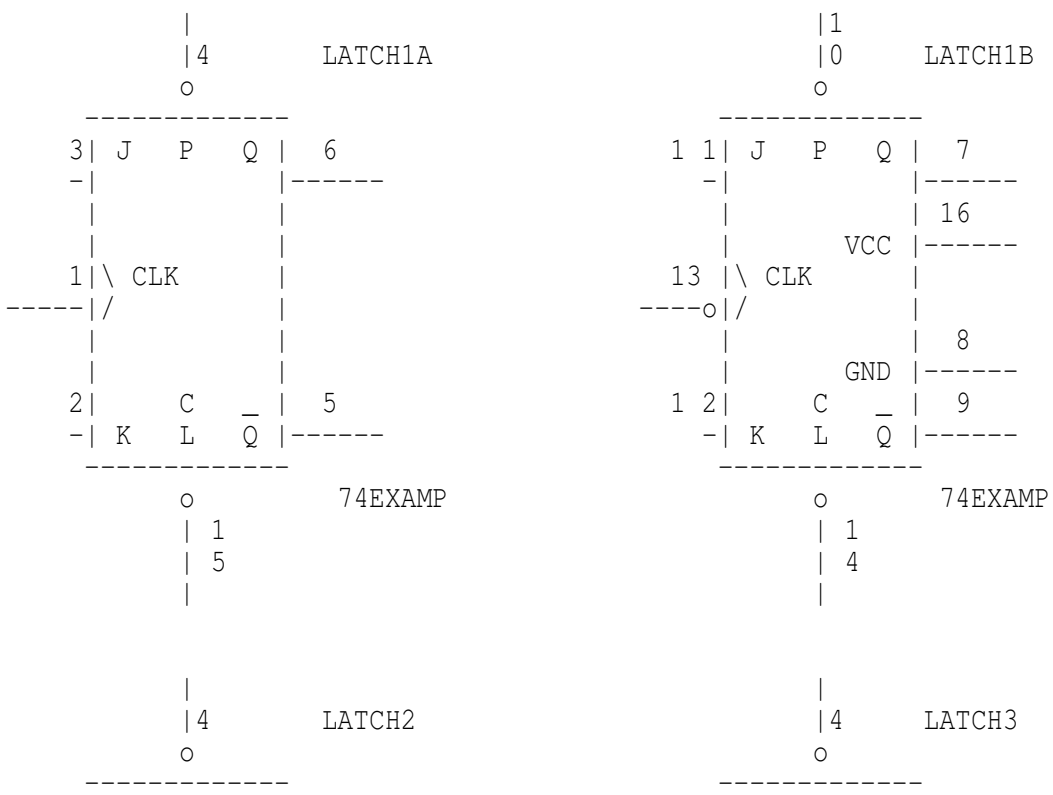


Figure 7-5A. Before Annotation

.pa
After Annotate



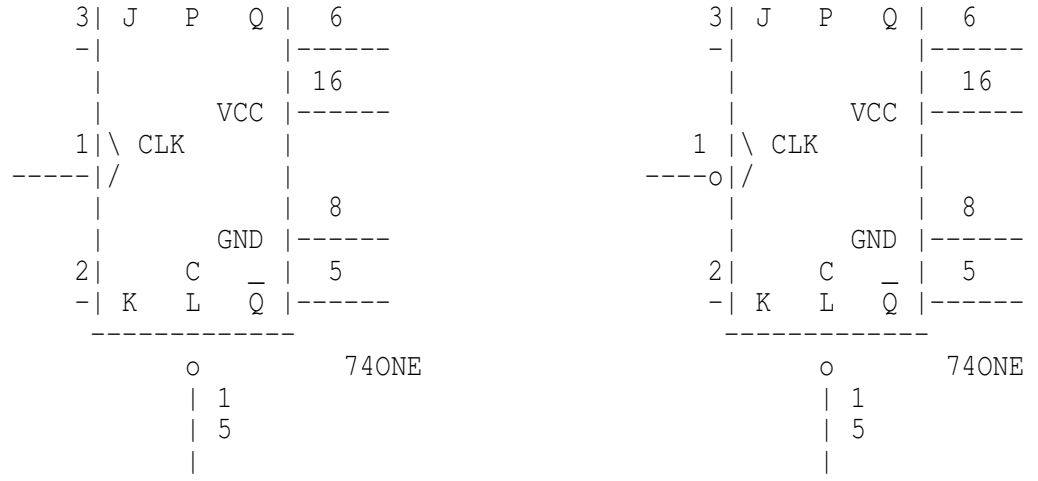


Figure 7-5B. After Annotation

If a device has more than one part per package, you may want power pins on some parts presents and the power pins on other parts not to be present.

Pin-grid array. If the part were a pin-grid array, you would supply the gridarray pin name instead of the pin number. A gridarray name consists of a capital letter followed by a number. The letter is in the range A through S, and the number is in the range 1 through 16. There can be no space between the letter and the number.

Here is an example of a pin-grid array part definition. The example is the 68020 from the Motorola library, MOTO.LIB. The definition is quite long so only the first few lines are shown.

Figure 7-6 shows the resulting screen figure.

```
'68020'
15   66   GRIDARRAY
L1   C2   CLK IN    'CLK IN'
L3   J12  IN        'I\P\L\0\'  (the \ bars the pin name)
L4   J13  IN        'I\P\L\1\'
L5   H12  IN        'I\P\L\2\'
```

(No Figure !)

Figure 7-6. The Block Symbol for 68020

Pin string. The pin string is delimited by single quotes. If you want a single quote as part of the pin string, you must use two single quotes. For example, 'CLK''s' defnes the string CLK's. Also, a backslash after the pin string name puts a bar over the name. 'Q\' results in Q with a bar over it. If you have a multi-letter pin name, you must put a \ after each letter. For example, the 68020 shows IPL0, and the corresponding pin string entry in the part definition is 'I\P\L\0'.

Note that the power pins do not appear on th screen. The NETLIST utility, however, does categorize all power supply pins that are connected to library parts.

7.3.4. Bitmap Symbol Definition

Creating a bitmap is an easy way to represent non-square or non-rectangular parts sucg as resistors, diodes, transistors, MOSFET's, realya, and many others. DRAFT draws complex parts on the screen by selectively tuming on pixel bits that represent the library part. Activating the correct pixel bit is controlled by a bitmap in the library source file you created. You create the bitmap in the library source file.

To define a part with a bitmap, you define the part just as you would if it were a block symbol, but you include a bitmap after the last pin definition. You can either draw out the bitmap with period (.)and pound signs (#), or you can reference a previously drawn-out bitmap. Previously drawn-out means the bitmap was defined previously in the library source file.

You would reference a previously drawn-out bitmap if two parts has different pinouts, but the same symbol. For example, the 7439 and the 7400 have the same symbol, but different pinouts. Assume that you've defined the 7400 and you're now defining the 7439. Instead of drawing another bitmap for 7439, you can use 7400's bitmap by including the line.

```
BITMAP '7400'
```

There are four points you should keep in mind when creating bitmaps, as opposed to block symbols.

1. You have to pay more attention to pin placement. The pin definition line and the bitmap have different scales,and you need to take the conversion into account when you draw the symbol.

2. Although you can put a pin name in the pin definition, the pin name will not appear on the screen. The pin name will, however, be recognized by the NETLIST utility.

3. A bitmap symbol gives you the opportunity to define a converted symbol. Bitmap devices always have a normal form. You have the option of also defining a converted form. Block symbols cannot have a converted form. When you use the GET command and extract a part from a library, it appears in normal form. The resulting menu enables you to choose its converted form instead. You define what the converted form is when you create the library source file. Typically, users define the converted form as the DeMorgan equivalent of the normal form.

4. The maximum number of bits allowed in a bitmap is 16.384. The bitmap begins after the last pin definition. A pound sign (#)indicates that the pixel bit is tumed on, and a period (.)indicates that the pixel bit is tumed off.

Then . or # represents a screen pixel spacing of 0.01 inch in the X direction. Each line of the bitmap represents 0.01 inch in the Y direction. Remember, the X and Y sizes in the part definition are given in units of 0.1 inch. For example, if you

specify X and Y to be 3 and 2, your bitmap actually is 31 characters in the X direction and 21 lines in the Y direction. The extra 1 results because the bitmap starts counting at zero.

An example should make this cleares. Here is a part definition for a resistor.

```
(Part definition for a resistor)
'resistor'
REFERENCE      'R'
3  2  0
L1   PAS       ' '
R1   PAS       ' '

.pa                          (Top side)
              (00).............................
              (01).............................
              (02).............................
              (03).............................
              (04).............................
              (05).............................
              (06).............................
              (07)...#...........#...........#...
              (08)..#.#.........#.#.........#.#..
              (09).#...#.......#...#.......#...#. (Right side)
              (10)#.....#.....#.....#.....#.....#
              (11).......#...#.......#...#.......
              (12).........#.#.........#.#........
              (13)..........#...........#.........
              (14).............................
              (15).............................
              (16).............................
              (17).............................
              (18).............................
              (19).............................
              (20).............................

                          (Bottom side)
```

Note that the number of parts per package is given as 0. Hence, there are no columns for pin numbers. The pin types are PAS for passive, and there are no pin names (however, they may be specified).

The X size is specified as 3, so the bitmap has lines that are 31 characters long. The Y size is specified as 2, so the bitmap has 21 lines. The part definition specifies two pins, one on the left in the first position (L1)and another in the right in the first position (R1). The pin positions are always spaced 0.1 inch apart. As far as the bitmap is concerned, pin positions are at lines 0, 10, 20, 30, etc.

The line number are enclosed in comment delimiters. It isn't necessary to number the bitmap lines this way, but doing so makes the bitmap more readable.

You can reduce the size of bitmapes used in your library by observing the following rules.

1. An empty row (one that has only dots)can be represented

by a dot in the zeroth column. If that is the only character on the row, then the row is held as cleared.

2. Empty rows below the actual symbol need not appear in the bitmap.

3. Periods are not required after the last # in a row.

Here is an example of the same resistot definition that follows the reduction rules just describes. Figure 7-7 shows the symbol that results from this part definition.

```
(Part definition for a resistor.)

'RESISTOR'
REFERENCE 'R'
3 2 0
L1   PAS  ' '
R1   PAS  ' '
```

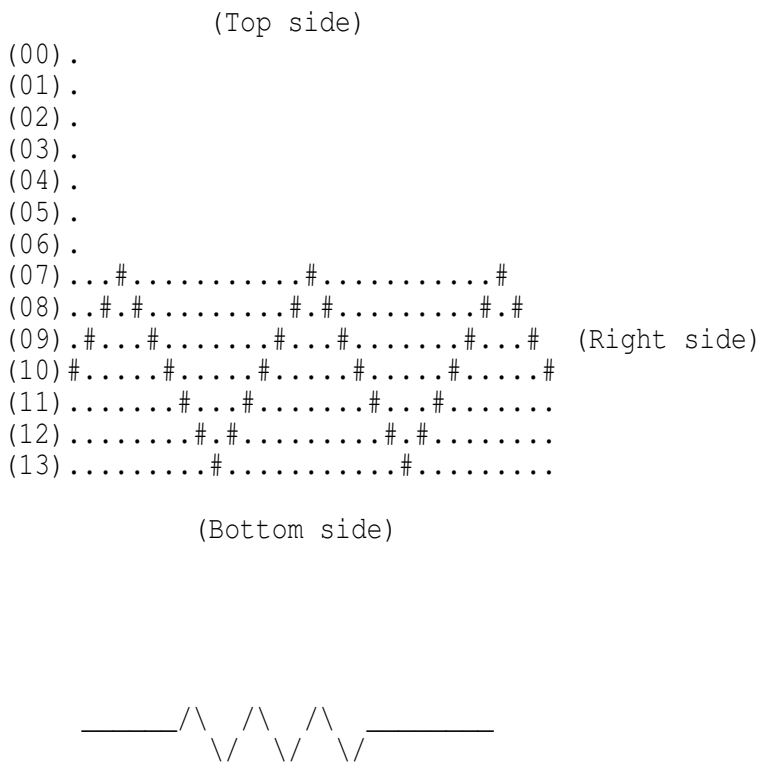


Figure 7-7. The Symbol for a Resistor

After defining a bitmap, you have the option of defining conversion bitmap. As stated previosly, the typical use of a conversion bitmap is to supply a DeMorgan equivalent symbol, but more generally a conversion specifies another bitmap that is displayed on the screen whenever you choose the Convert option of the GET command. You can return to the original bitmap by selecting the Normal subcommand option.

You begin the specification of a conversion bitmap with the keyword CONVERT. The conversion bitmap consist of a pin definitions followed by a bitmap. If the conversion has been

previously defined, you can reference it by including the name of the part that has the conversion in single quotes.

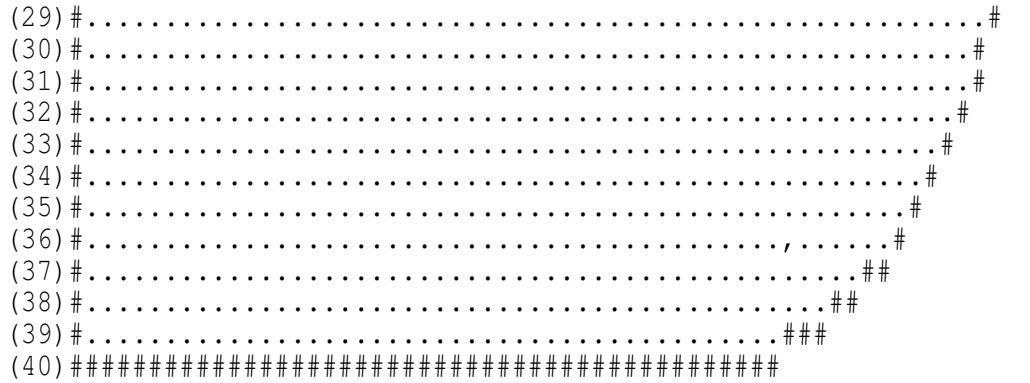
The next example should clarify the use of conversion bitmap. First, is the definition of the 7400. Figure 7-8 shows the normal and convrted symbols that results from this definition.

The 7400 has five pins, two of which are power pins that do not appear in the symbol. The screen size is 6 X-units and 4 Y-units. It has four parts per package.

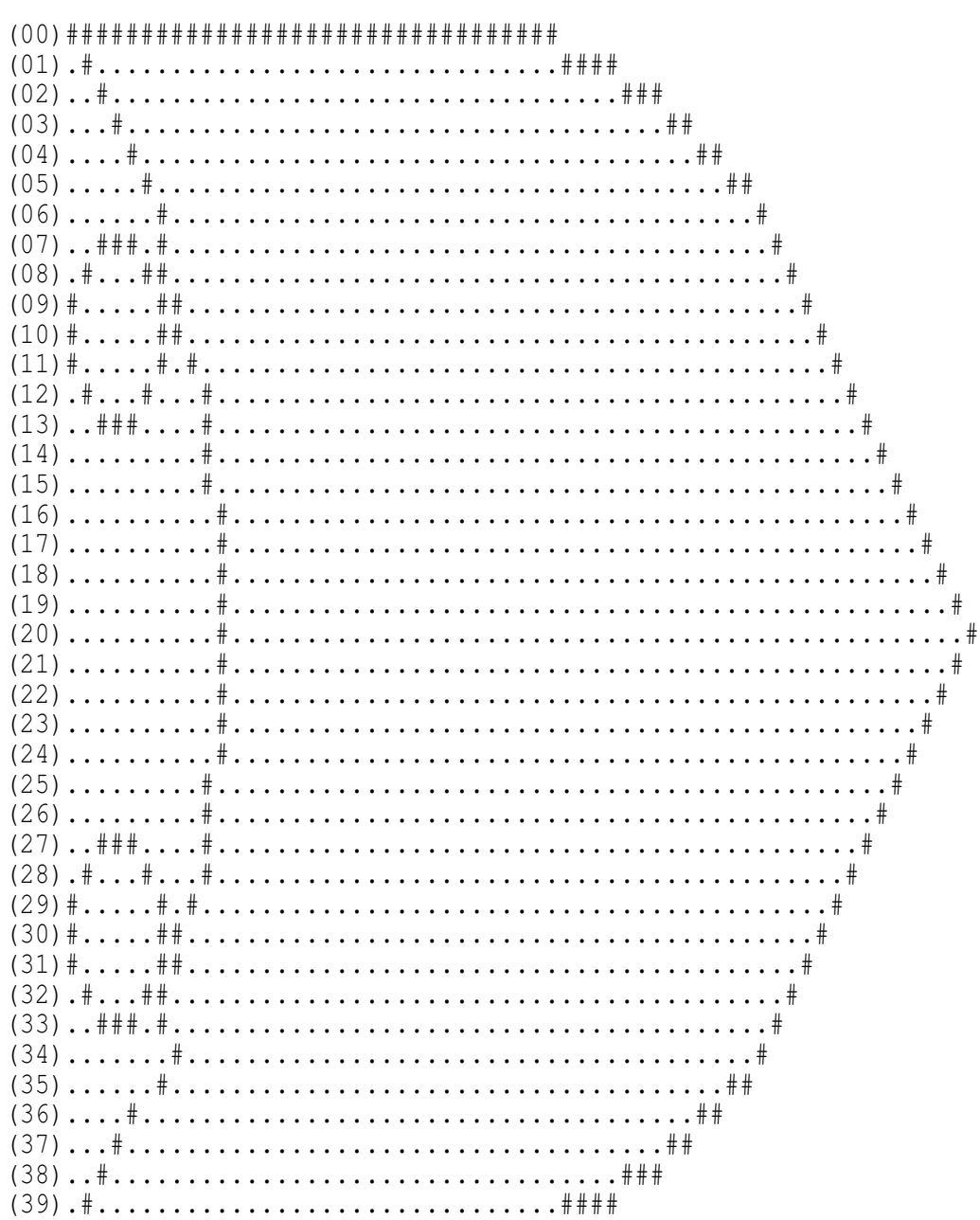
The conversion bitmap uses the same XY size and parts per package as the normal bitmap. You must, however, redefine the pins. Note the DOT keyword missing from the redefinition of the pin at R2.

Note that the conversion bitmap has the same number of parts per package as the normal bitmap. The number of parts per package determines how many columns of pin numbers appear in the definition. The converted definition must have the same number of columns as the normal definition.

```
.pa
'7400'
6    4    4
L1   1    4    9    12   IN   'I0'
L3   2    5    10   13   IN   'I1'
R2   3    6    8    11   DOT  OUT  'O'
T0   14   14   14   14        PWR  'VCC'
B0   7    7    7    7         PWR  'GND'

(00)##############################################
(01)#.............................................###
(02)#...........................................##
(03)#.........................................##
(04)#.......................................#
(05)#.....................................#
(06)#...................................#
(07)#.................................#
(08)#...............................#
(09)#.............................#
(10)#...........................#
(11)#.........................#
(12)#.......................#
(13)#.....................#
(14)#...................#
(15)#.................#
(16)#...............#
(17)#.............#
(18)#...........#
(19)#.........#
(20)#.......#
(21)#.....#
(22)#...#
(23)#.#
(24)#.#
(25)#...#
(26)#.....#
(27)#.......#
(28)#.........#
```



```
(29)#.....................................................#
(30)#.....................................................#
(31)#.....................................................#
(32)#....................................................#
(33)#....................................................#
(34)#...................................................#
(35)#..................................................#
(36)#.............................................,.....#
(37)#.................................................##
(38)#...............................................##
(39)#............................................###
(40)##########################################
.PA
CONVERT
L1   1    4    9    12        IN   'I0'
L3   2    5    10   13        IN   'I1'
R2   3    6    8    11        OUT  'O'
T0   14   14   14   14        PWR  'VCC'
B0   7    7    7    7         PWR  'GND'
```



```
(00)################################
(01).#.............................####
(02)..#...........................###
(03)...#.........................##
(04)....#.......................##
(05).....#.....................##
(06)......#....................#
(07)..###.#....................#
(08).#...##....................#
(09)#.....##...................#
(10)#.....##...................#
(11)#.....#.#..................#
(12).#...#...#.................#
(13)..###....#.................#
(14).........#.................#
(15).........#.................#
(16)..........#................#
(17)..........#................#
(18)..........#................#
(19)..........#................#
(20)..........#................#
(21)..........#................#
(22)..........#................#
(23)..........#................#
(24)..........#................#
(25).........#.................#
(26).........#.................#
(27)..###....#.................#
(28).#...#...#.................#
(29)#.....#.#..................#
(30)#.....##...................#
(31)#.....##...................#
(32).#...##....................#
(33)..###.#....................#
(34).......#...................#
(35)......#....................##
(36)....#.....................##
(37)...#.....................##
(38)..#.....................###
(39).#.............................####
```

(40)################################
.pa

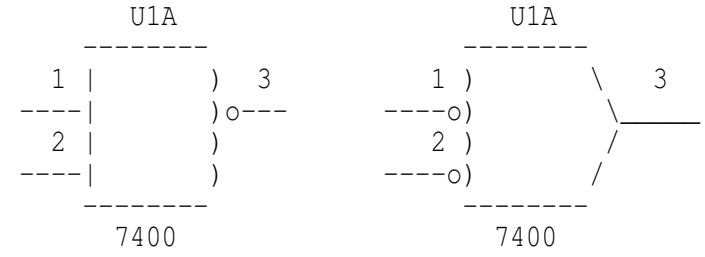


Figure 7-8. The 7400 Symbol and its Conversion

7.4. SDL Reference

This section is a complete description of how to define a part in a custom library. It presents OrCAD's Symbol Description Language (SDL) in the form of syntax diagrams. A syntax diagram consists of identifiers (enclosed in ovals) and tokens (enclosed in rectangles). The syntax diagram is also followed by a textual representation.

7.4.1. Syntax Diagram

Figure 7-9 is an example of a syntax diagram. It represents the complete syntax for a library source file.

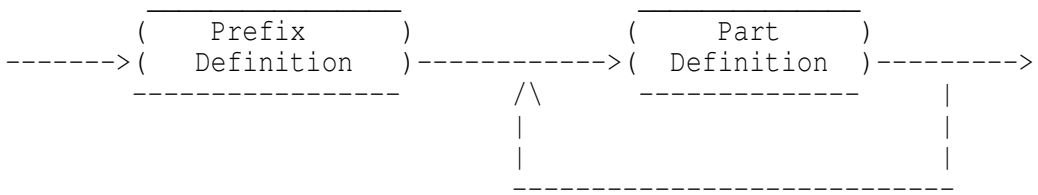


Figure 7-9. Syntax Diagram for a Library Source File

To read a syntax diagram, observe the following rules.

1. Read a syntax diagram from left to right.

2. Every path that you follow in the direction of the arrows represents a correct syntax form.

3. Junctions represents a connection point where you have the choise of selecting another path. For example, the syntax diagram for the library source file shown in Figure 7-9 has a junction after a part definition. You can choose to complete the diagram or to make another part definition.

4. You cannot continue along a path that goes against the arrows. For example, in Figure 7-9 after making a part definition you cannot choose to return to the prefix definition.

5. Text enclosed in ovals represents an identifier. Text

enclosed in squares represents a token.

What are identifiers? Identifiers serve as placeholders for a more detailed level of syntax structure. They do not represent command syntax or tokens. Rather, they provide the ability to give a overview of the syntax. When you create the part, you must work down through all the nested identifiers. For example the syntax diagram for a library source file shown in Figure 7-9 has two identifiers and no tokens. This section will now axpand each of those identifiers and present all the possible tokens.

What are tokens? They are the building blocks of a library source file. Just as a sentence is made up of words, a library source file is made up of tokens. A token belongs to one of the following categories.

.PN202
.HE Schematic Design Tools Libraries

What are tokens? They are the building blocks of a library source file. Just as a sentence is made up of words, a library source file is made up of tokens. A token belongs to one of the following categories.

* Numeric constants. A numeric constant consists of one or more whole-number digits.

  Examples: 15
  2
  127

* Character strings. A character string consists of one or more alphanumeric ASCII characters.

  Example: 74ALS04
  ZENER
  L5
  CLOCK

* Keywords. A keyword is one of the following:

  BITMAP Takes an argumment (an ASCII string representing a part name) and represents the bitmap of the identified part.

  CLK Represents the clock symbol in a pin definition.

  CONVERT Introduces a converted bitmap. With an argument, it refers to the converted bitmap of a bitmap symbol.

  DOT Represents the inversion bubble in a pin definition.

  END Delimits the close of a prefix definition.

  GRIDARRAY Specifies that the device is a pin-grid array. Used in place of the number of parts per package.

HIZ	Identifies the pin as a high impedance (3-state) output.
IN	Identifies the pin as an input.
I/O	Identifies the pin as input/output.
OC	Identifies the pin as open collector.
OUT	Identifies the pin as an output.
PAS	Identifies the pin as passive.

.PA

PREFIX	Delimits the beginning of a prefix definition.
PWR	Identifies the pin as a power pin. The PWR keyword prevents a pin from being displayed.
REFERENCE	Takes an argument (an ASCII string representing a reference value). Overrides the default reference value.
SHORT	Specifies that the pin lead lengths be 0.1 inch instead of the normal 0.3 inch.

7.4.2. Textual Representation of Syntax

In addition to the syntax diagram, syntax is represented in text. The symbols are defined as follows.

Text enclosed in italics represents either a character string a numeric constant.

[] Text enclosed in square brackets is optional. You choose whether to type it in or not. Do not type the square brackets.

{} Text enclosed in braces is required. You must enter what's represented within the braces.

, If items within square brackets or braces are separated by commas, you must choose one of them only. Do not type the comma.

... Three periods mean you can repeat the last item. How many times you can repeat the item depends on the context. Do not type the periods.

Here is an example of syntax represented in text.

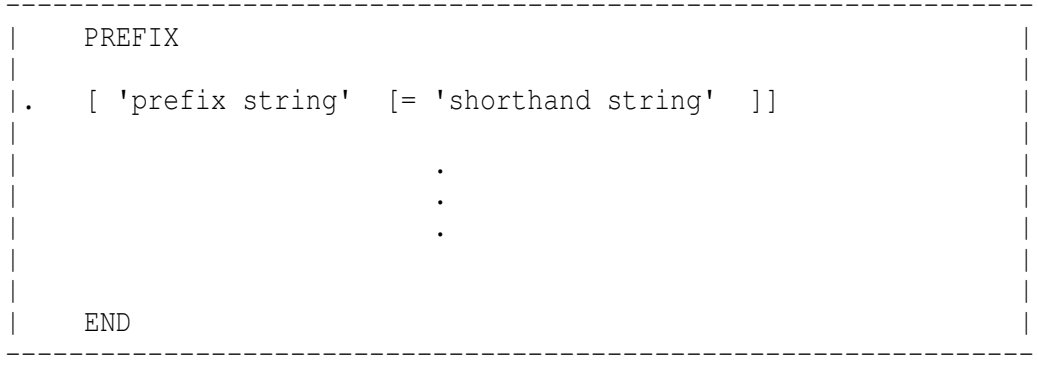
pos [pin# .... grid ...]

First you must enter a representation of pos. An accompanying description explains what you can put in for pos. The square brackets around the text item indicate that you don't have to enter it at all. If you do, you must choose between pin# or grid; and you can choose a repetead number of each.

.pa

7.4.3. PREFIX Definition

Figure 7-10. Syntax Diagram For a Prefix Definition



where:

prefix string	A character string of up to seven printable ASCII characters. You can have a maximum of 16 prefix strings.
shorthand string	A character string of up to seven printable ASCII characters.

NOTES

The equal sign must be separated by one or more <SPACE>s or <TAB>s.

EXAMPLE

Example:

```
PREFIX
'74LS'          =  'LS'
'74S'           =  'S'
'74ALS'         =  'ALS'
'74AS'          =  'AS'
74HCT'          =  'HCT'
'74HC'          =  'HC'
'74ACT'         =  'ACT'
'74AC'          =  'AC'
'74F'           =  'F'
'74'
END
```

Example:

```
PREFIX
END
```

7.5.4. Part Definition

Figure 7-11. Syntax Diagram for a part Definition

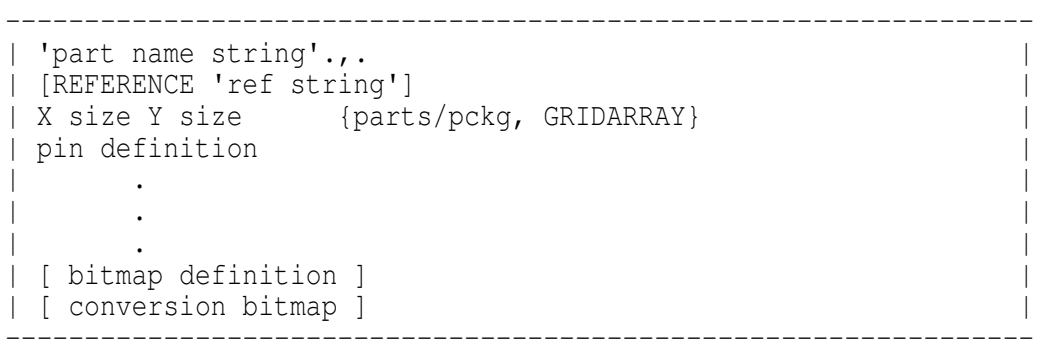


Figure 7-12. The Part Name String

where:

part name string	A character string of ut to seven printable ASCII characters that identifies the part. This is the string that can be used as an argument for the Get command.
ref string	A character string of printable ASCII characters. If present, the reference designator replaces the default eference designator.
X size	A numeric constant in the range 1 to 127. The horizontal size of the part at it appears on a printed worksheet. Each entry corresponds to one unit length on the screen or 0.1 inch on a printed worksheet.
Y size	A numeric constant in the range 1 to 127. The vertical size of the part as it appears on a printed worksheet. Each entry corresponds to one unit lenght on the screen or 0.1 inch on the printed worksheet.
parts/pckg	A numeric constant in the range 0 to 16. If you specify a 0, the pins are not numbered on the symbol.
pin definition	These are identifiers for a more detailed bitmap definition level osyntax. See the corresponding conversion bitmap entry.

NOTES

To improve readability, you can include comments within a part definition. You can also place blank lines within a source file. Typically blank lines are placed between different part definition.

EXAMPLES

```
Example 1:     '2114'    '2148'    {Two  part name  strings.
                6     14     1     These  may be on the same
                                   or separate lines.}
                 pin definition
                    .
                    .
                    .
```

Example 2:

```
     '7474'    '74ALS74'    '74LS74'    '74S74'
     '74HC74'  '74AC74'
     6     6      2
     pin definition
          .
          .
          .
```

7.4.5. Pin Definition

Figure 7-13. Syntax Diagram for a Pin Definition

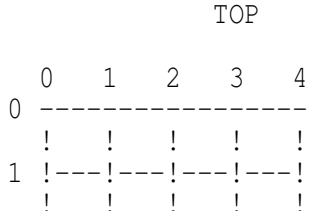
pos	[pin#,grid]	[DOT] [CLK] [SHORT]	[IN] [OUT] [I/O] [OC] [PWR] [PAS] [HIZ]	pin name

where:

pos — A letter followed by a number. The letter is one of the following T, L, R, B, where

T indicates the top of the symbol.
L indicates the left side of the symbol.
R indicates the right side of the symbol.
B indicates the bottom of the symbol.

The nimber represents the distance along the indicated side. The distance is measured in unit lengths on the screen and in 0.1 inches on the printed worksheet. For example, if the block symbol were 6X by 10Y the grid used for placing pins is as follows. The figure below shows the location of L3.



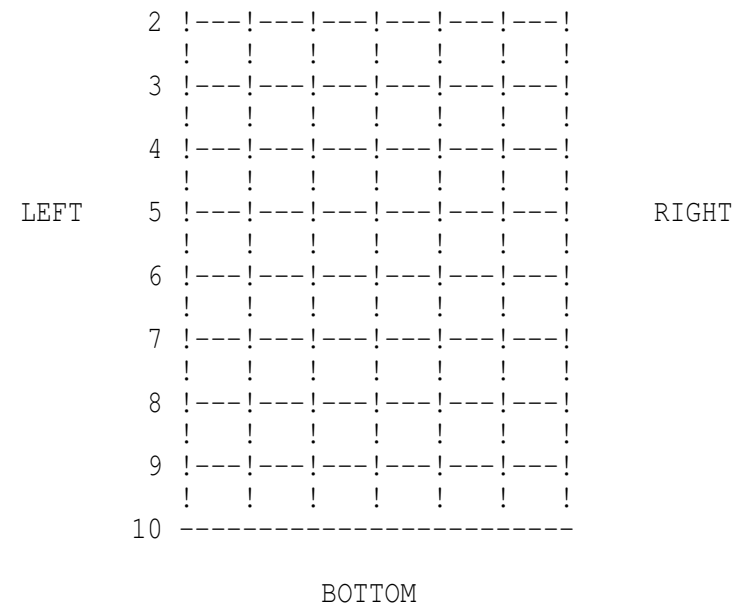


Figure 7-14. The grid of a 6X by 10Y Block Symbol

(No Figure !)

Figure 7-15. The Position Syntax Diagram

where:

pin# — A numeric constant representing the pin number. This is the pin number that appears in the screen symbol.

pin grid — A letter followed by a number, grid represents the pin-grid array pin number. Ypu can only choose a pin-grid array pin number if, in place of parts/pckg, you choose the key word GRIDARRAY. The letter must be in the range A through S; the number must be in the range 1 through 15.

GRID ARRAY PIN NAME

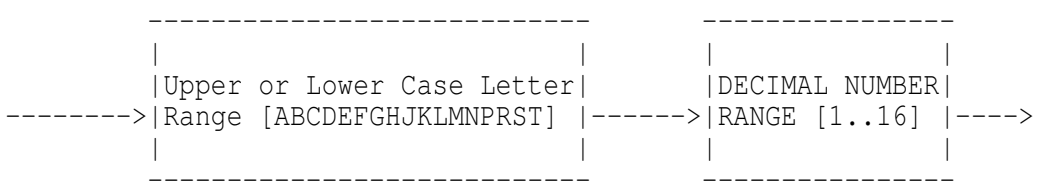


Figure 7-16. Gridarray Pin Name Syntax Diagram

SHORT — A keyword that places a short lead length at the specified pin. The normal lead length is

3 screen units or 0.3 inches on the printed worksheet. When the SHORT keyword is present, the lead length is 1 sceen unit or 0.1 inch on the printed worksheet. The SHORT keyword cannot describe a pin that also has either the CLK or DOT keywords.

DOT A keyword that places then inversion symbol (the bubble) at the specified pin location. The DOT keyword cannot describe a pin that also has the SHORT keyword. The primary use of the bubble is identify pins that have logic bubble is to identify pins that have logic negation, either at an input or an at output. The fifure below shows the inversion symbol.

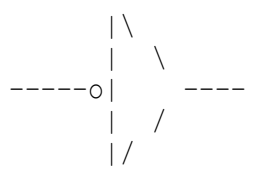


Figure 7-17. The Inversion DOT Symbol

CLK A keyword that places the clock symbol at the specified pin location. The CLK keyword cannot describe a pin that also has the SHORT keyword. The figure below shows the CLK symbol.

.pa

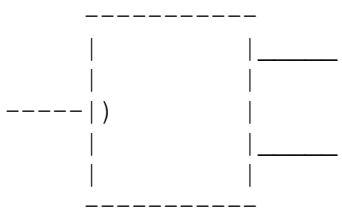


Figure 7-18 A. The CLK Symbol

You can use the CLK keyword in conjunction with the DOT keyword to produce a DOT CLK symbol. The figure below shows the DOT CLK symbol.

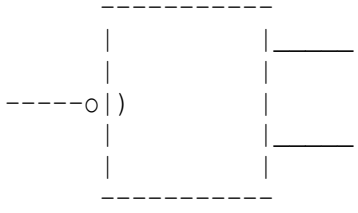


Figure 7-18 B. The DOT CLK Symbol

IN A keyword that identifies the pin as an input.

OUT A keyword that identifies the pin as a

standard totem-pole output.

I/O	A keyword that identifies the pin as a dual function input/output pin.
OC	A keyword that identifies the pin as an open collector or open drain.
PWR	A keyword that identifies a power pin, such as Vcc, Gnd, Vss, and others. Power pins are not displayed on library parts when they appear in the screen or printed worksheet. However, the NETLOST utility connects all power supply pins that are defined in library source files.
PAS	A keyword that identifies a pin as passive. Passive pin are typically pins on passive devices such as resistors, transistors, inductors, and others.
HIZ	A keyword that identifies a pin as a high-impedance (3-state) output.
pin name	A character string that represents a name for the specified pin. For block symbols, this name appears on the screen or the printed worksheet. Pin names do not appear on the screen or the printed worksheet when they are part of pin definition for a bitmap symbol. However, you may still choose to use pin names in bitmap symbols. The NETLIST utility still recognizes them, and you may find them useful as personal references.

7.6.1 SAMPLE - 1.SRC:A Block Symbol Library Source File

```
{Beginning of source file, SAMPLE-1.SRC}

PREFIX
END
{THE following are memory devices}

{1K x 4 static RAM}
'2114'  '2148'  '2149'  '5513'  '5514'  '9114'  '9124'
6       14      1
L1      5       IN      'A0'
L2      6       IN      'A1'
L3      7       IN      'A3'
L4      4       IN      'A4'
L5      3       IN      'A5'
L6      2       IN      'A6'
L7      1       IN      'A7'
L8      17      IN      'A7'
L9      16      IN      'A8'
L10     15      IN      'A9'
L12     8       IN      'C\S\'
```

```
L13     10       IN     'W\E\'
R1      14       HIZ    'D0'
R2      13       HIZ    'D1'
R3      12       HIZ    'D2'
R4      11       HIZ    'D3'
T0      18       PWR    'VCC'
B0      9        PWR    'GND'

{All of the above memory devices have the same block symbol.  The
part's size is 6X by 14Y, and it has one part per package.}

{256K x 1 dynamic RAM}
'21256'  '51C256'  '50256'  '50257'  '81256'  '81257'  '6256'
'6257'   '4256'    '4257'   '41256'  '37256'
6       14       1
L1      5        IN     'A0'
L2      7        IN     'A1'
L3      6        IN     'A2'
L4      12       IN     'A3'
L5      11       IN     'A4'
L6      10       IN     'A5'
L7      13       IN     'A6'
L8      9        IN     'A7'
L9      1        IN     'A8'
L11     4        IN     'R\A\S'
L12     15       IN     'C\A\S'
L13     3        IN     'W\E\'
R1      14       HIZ    'D0'
R3      2        IN     'D1'
T0      8        PWR    'VCC'
B0      16       PWR    'VSS'

{4K x 8 EPROM}
'2732'
7       16       1
L1      8        IN     'A0'
L2      7        IN     'A1'
L3      6        IN     'A2'
L4      5        IN     'A3'
L5      4        IN     'A4'
L6      3        IN     'A5'
L7      2        IN     'A6'
L8      1        IN     'A7'
L9      23       IN     'A8'
L10     22       IN     'A9'
L11     19       IN     'A10'
L12     21       IN     'A11'
L14     18       IN     'C\E\'
L15     20       IN     'O\E\/VPP'
R1      9        HIZ    'O0'
R2      10       HIZ    'O1'
R3      11       HIZ    'O2'
R4      13       HIZ    'O3'
R5      14       HIZ    'O4'
R6      15       HIZ    'O5'
R7      16       HIZ    'O6'
R8      17       HIZ    'O7'
T0      24       PWR    'VCC'
B0      12       PWR    'GND'
```

```
{The following are microprocessor devices}

'8086MAX'
13      32       1
R1      16       I/O    'AD0'
R2      15       I/O    'AD1'
R3      14       I/O    'AD2'
R4      13       I/O    'AD3'
R5      12       I/O    'AD4'
R6      11       I/O    'AD5'
R7      10       I/O    'AD6'
R8      9        I/O    'AD7'
R9      8        I/O    'AD8'
R10     7        I/O    'AD9'
R11     6        I/O    'AD10'
R12     5        I/O    'AD11'
R13     4        I/O    'AD12'
R14     3        I/O    'AD13'
R15     2        I/O    'AD14'
R16     39       I/O    'AD15'
R17     38       OUT    'A16/S3'
R18     37       OUT    'A17/S4'
R19     36       OUT    'A18/S5'
R20     35       OUT    'A19/S6'
R22     34       OUT    'B\H\E\/S7'
R24     26 DOT   OUT    'S0'
R25     27 DOT   OUT    'S1'
R26     28 DOT   OUT    'S2'
R28     32       OUT    'R\D\'
R29     29 DOT   OUT    'LOCK'
R30     25       OUT    'QS0'
R31     24       OUT    'QS1'
L3      22       IN     'READY'
L4      19       CLK IN 'CLK'
L5      21       IN     'RESET'
L7      18       IN     'INTR'
L27     31 DOT   I/O    'RQ/GT0'
L28     30 DOT   I/O    'RQ/GT1'
L29     17       IN     'NMI'
L30     23 DOT   IN     'TEST'
L31     33       IN     'M\X\'
T0      40       PWR    'VCC'
B0      20       PWR    'GND'
B5      1        PWR    'GND'

{This devices is in a pin-grid array package.  Notice the keyword
GRIDARRAY in place of parts per package}

'68020'
16      53       GRINDARRAY
L1      C2                CLK IN   'CLK'
L3      J12 DOT           IN       'IPL0'
L4      J13 DOT           IN       'IPL1'
L5      H12 DOT           IN       'IPL2'
L7      H2  DOT           IN       'AVEC'
L8      A1  DOT           IN       'BGACK'
L9      B3  DOT           IN       'BR'
L10     J2  DOT           IN       'BERR'
L11     H1  DOT           IN       'CDIS'
L13     H3  DOT           IN       'DSACK0'
```

L14	J1 DOT	IN	'DSACK1'
L21	K13	I/O	'D0'
L22	K12	I/O	'D1'
L23	L13	I/O	'D2'
L24	L12	I/O	'D3'
L25	M13	I/O	'D4'
L26	M12	I/O	'D5'
L27	M11	I/O	'D6'
L28	L10	I/O	'D7'
L29	N12	I/O	'D8'
L30	N11	I/O	'D9'
L31	M10	I/O	'D10'
L32	L9	I/O	'D11'
L33	N10	I/O	'D12'
L34	M9	I/O	'D13'
L35	N9	I/O	'D14'
L36	L8	I/O	'D15'
L37	M7	I/O	'D16'
L38	N6	I/O	'D17'
L39	M6	I/O	'D18'
L40	L6	I/O	'D19'
L41	N5	I/O	'D20'
L42	M5	I/O	'D21'
L43	N4	I/O	'D22'
L44	L5	I/O	'D23'
L45	M4	I/O	'D24'
L46	N3	I/O	'D25'
L47	M3	I/O	'D26'
L48	L4	I/O	'D27'
L49	N2	I/O	'D28'
L50	M2	I/O	'D29'
L51	L3	I/O	'D30'
L52	N1	I/O	'D31'
R1	C4	OUT	'A0'
R2	A2	OUT	'A1'
R3	E12	OUT	'A2'
R4	D13	OUT	'A3'
R5	D12	OUT	'A4'
R6	C13	OUT	'A5'
R7	B13	OUT	'A6'
R8	C12	OUT	'A7'
R9	A13	OUT	'A8'
R10	C11	OUT	'A9'
R11	B12	OUT	'A10'
R12	A12	OUT	'A11'
R13	C10	OUT	'A12'
R14	B11	OUT	'A13'
R15	A11	OUT	'A14'
R16	B10	OUT	'A15'
R17	C9	OUT	'A16'
R18	C8	OUT	'A17'
R19	B8	OUT	'A18'
R20	A8	OUT	'A19'
R21	B7	OUT	'A20'
R22	C7	OUT	'A21'
R23	A7	OUT	'A22'
R24	A6	OUT	'A23'
R25	B6	OUT	'A24'
R26	C6	OUT	'A25'
R27	A5	OUT	'A26'

```
R28     B5              OUT     'A27'
R29     A4              OUT     'A28'
R30     C5              OUT     'A29'
R31     B4              OUT     'A30'
R32     A3              OUT     'A31'
R34     F13 DOT         OUT     'IPEND'
R35     B2  DOT         OUT     'BG'
R37     E1              OUT     'FC0'
R38     F3              OUT     'FC1'
R39     F2              OUT     'FC2'
R41     F1              OUT     'SIZ0'
R42     G2              OUT     'SIZ1'
R43     G3  DOT         OUT     'DBEN'
R44     G1  DOT         OUT     'ECS'
R45     E13 DOT         OUT     'OCS'
R46     E2  DOT         OUT     'RMC'
R47     L1  DOT         OUT     'AS'
R48     M1  DOT         OUT     'DS'
R50     L2              OUT     'R/W\'
R51     K2  DOT         I/O     'HALT'
R52     C1  DOT         I/O     'RESET'
T0      A9              PWR     'VCC'
T1      D1              PWR     'VCC'
T3      D2              PWR     'VCC'
T6      E3              PWR     'VCC'
T9      G11             PWR     'VCC'
T12     G13             PWR     'VCC'
T15     M8              PWR     'VCC'
T16     N8              PWR     'VCC'
B0      A10             PWR     'GND'
B1      G12             PWR     'GND'
B3      H13             PWR     'GND'
B6      J3              PWR     'GND'
B9      K1              PWR     'GND'
B12     L7              PWR     'GND'
B15     N7              PWR     'GND'
B16     B9              PWR     'GND'

{End of source file, SAMPLE-1.SRC}
.PN223
.he Tutorial                            Schematic Design Tools
```

8. TUTORIAL

This section is a step-by-step tutorial that shows how to create a schematic worksheet and take advantage of the OrCAD utilities. The tutorial creates a simple one-sheet schematic for a portion of a memory-decodate circuit. The circuit is similar to that found in the IBM PC and compatible computers.

This section assumes that you have configured OrCAD/SDT. That is, you have invoked DRAFT with the configuration option (/C) and specified the directory that contains the display and printer drivers, the directory that contains the library files, the directory that will contain the worksheets, and the pathname of an aoptional macro file. You have also added the names of the library files you intend to use in your design. These names may include custom libraries that you have written.

The section is divided into three modules. Each module contains step-by-step instructions, set off from the text with the identifier Action: If you follow those instructions, you'll end up with a one-sheet schematic of the memory-decode circuit.

Figure 8-1 shows the worksheet at the conclusion of Module 1, and Figure 8-2 shows the worksheet at the end of Module 2.

The acompanying text describes the instructions in more detail and ocasionally presents alternate methods. The intent is to demonstrate the power of OrCAD/SDT while still keeping the example simple and straightforward.

8.1. Assumptions

Hard Disk. When describing the directory structure recommended for hard disk users, this section assumes that you have configured DOS to show the directory structure as part of your prompt. For example, the default prompt for drive c is C>. If the directory C:\orcad is your working directory, this section assumes that your prompt is C:\orcad>. To configure DOS this way, issue the DOS command.

```
C>PROMPT$P$G
```

If you prefer, include this line in the file autoexec.bat and place it in the root directory of your boot device so that the configuration takes place automatically every time you boot up.

Mouse. Although not required, a mouse makes life much easier. This example assumes that you have a mouse with its driver installed.

OrCAD/SDT makes use of the left and right mouse buttons. If your mouse has a middle button, you may define it as a macro key. The left mouse button executes <ENTER> and the right mouse button executes <ESCAPE>.

If you are not using a mouse, you can execute <ENTER> by pressing the <RETURN> key and <ESCAPE> by pressing the <ESC> key. Typing <RETURN> after invocation brings up the main command menu. You can then execute a command by typing the first letter of the command name. You can also highlight the command to be executed by moving the highlight with the directional arrow keys and the choosing the highlighted command by pressing <Return>.

Printer. This tutorial assumes that you have an attached printer. That is, you have configured OrCAD/SDT with the appropriate printer driver and that your printer is connected to the first parallel, LPT1.

You can also use the PRINTALL utility to print your schematics. If you are using a plo%ter instead of a printer, you cannot use the hardcopy command. The PLOTALL utility is used to plot your schematics.

8.2. Requirements

The example requires that you have configured OrCAD/SDT with the following libraries.

Module 1: TTL.LIB
Module 2: TTL.LIB,DEVICE.LIB, a custom library

The custom library in Module 2 consists of one part. The module describes how to create the custom library.

8.3. Module 1: Constructing a Worksheet, Part 1

Summary: This module consists of invoking DRAFT, setting its

environment, extracting some parts from the appropriate libraries, editing the reference block, saving the work sheet, and making a hardcopy.

Figure 8-1. The worksheet after Performing Module 1

8.3.1. Invoking DRAFT

When you first begin your design, you must specify a worksheet. Note that you need not specify the complete pathname of the worksheet. If you specify only a filename, DRAFT assumes that your works,eet resides in the directory that you specified as the worksheet prefix during configuration. You can, of course, override the default you supplied during configuration by giving a complete pathname.

Specify a worksheet in one of three ways. The following examples assume that tuto.sch is the name of your worksheet.

1. Include the name of your worksheet on the invocation line. If the file already exists, DRAFT brings it in for editing. If the file does not exist, DRAFT uses the name you supplied when you update or write the file. If you abandon without editing (that is, choose the Abandon Edits command without having update or written the file), the file will not exist.

```
CORCAD>DRAFT TUTOR.SCH
```

2. Specify the name of your work file when DRAFT requests a load file. If you invoke DRAFT without a worksheet, you receive the prompt load file? after the company logo and the copyright message. Respond with the name of your worksheet.

```
C:\ORCAD>DRAFT

OrCAD logo appears; type any key to continue.>
<Copyright information appears; type any key to continue.>

load file?tutor.sch
```

3. Specify the name of the file when you update it or write it. If you invoke DRAFT without a worksheet and respond with an <ENTER> when DRAFT requests a load file, DRAFT assumes you are working with an unnamed worksheet. If you choose to write that unnamed worksheet to file, DRAFT requests the name of the file.

```
C:\ORCAD>DRAFT

<Respond with <ENTER> when load file? appears. Construct the
file. Choose the QUIT command. Choose the Write command.>

Write to file?tutor.sch
```

Action: Invocation. Enter the OrCAD directory. This tutorial assumes that directory is C:\ORCAD. Issue the command.

```
C:\ORCAD>DRAFT
```

When the logo appears, type any key. Then, when the copyright information appears, also type any key. Then, when the prompt load file? appears, type tutor.sch followed by an <ENTER>.

8.3.2. Setting the DRAFT Environment

You are now in DRAFT. You see the top and left lines that define the sheet. On the prompt line is the message <<New Worksheet>>. Because the screen is smaller than the sheet size, the other two defining ? (the right and bottom) are off screen. You can think of the screen as a window into the worksheet.

DRAFT is invoked with a default environment. To see what that environment is, issue the SET command.

Action: Looking at Environment Settings. Press the left button on your mouse. The main command menu appears. Select [he SET command. That is, move the mouse to highlight the SET command and press the left button. The SET menu appears.

How yoy set DRAFT's environment is often a matter of personal preference. For example, Auto Pan's default is YES. This means that when you move the mouse past the edge of the screen, the screen follows.

Action: Panning across the Worksheet. First get rid of the menu by pressing the right button on your mouse. This is equivalent to pressing the <ESC> key. Then, move the mouse to the lower right corner and keep moving it until the title block appears. The screent pans to keep up with your mouse.If you set Auto Pan to NO, the screent will not pan: but you can still get to different parts of your sheet by issuing the JUMP command.Now move the mouse to the upper left hand corner, returning the screen to its original location on the work sheet.
Press the left button. The main command menu appears. The highlight is on AGAIN. Choose AGAIN by pressing the left button. This executes the last command. Because the command was SET, the SET menu reappears.

More SET Options

Backup File determines whether or not a backup file is produced. With this option set, your disk will have two work files when you exit DRAFT, tutor.sch (the file you created) and tutor.bak (the backup file). The backup file is the resulting file from your last edit session. If you're editing a brand new file (as in this example), you, of course, won't have a backup. Unless your disk space is severely limited, you should let Backup File remain at YES.

Drag Buses enables the rubberbanding of buses. This is in addition to rubberbanding of wires, which is always enabled. Rubberbanding buses is a feature you should use only when necessary. DRAFT has to keep track of many more points when dragging buses, and you may notice a decrease in performance.

Error Bell determines whether or not your computer's bell sounds when you try to do something illegal. How you set this option is a matter o fpersonal preference.

Be sure to leave Left Button at NO. Althought setting Left Button to YES does speed up some mouse commands, this tutorial assumes Left Button is NO.

With Left Button set to YES, you can skip a button press when executing some mouse commands. For example, you can call up a menu by pressing the left button and, while being careful to keeep the left button down, highlight the command you want to execute by moving the mouse. You can then execute the command by releasing the button. Contrast this with pressing the left button to get the menu, highlighting the command, and pressing the button again to execute the highlighted command.

After you become experienced with macros, you may want to set Macro Prompts to NO. Leaving it at YES gives you more feedback about what's hoing on when you execute a macro.

Setting Orthogonal to NO, allows you to draw wires at arbitrary angles. By default, wires are drawn orthogonally (that is they are drawn either horizontally or vertically), and turns are always right lines.

Show Pin determines whether or not the pin numbers on devices are shown. If Title Block is NO, the title block in the lower right hand corner is disabled. This is the same title block you looked at when you tried out autopannig. You may choose to disable this default title block and construct one of your own.

Leave Worksheet Size set at A. This restricts your worksheet to 8.5. by 11 inches. The actual working area is, of course, somewhat smaller. X, Y Display defults as NO. Change it to YES.

Action: Setting X, Y Display to YES. Move the mouse so that the highlight is on X, Y Display. press the leftt button again. Move the mouse in any direction and observe the XY coordinates in the upper rightland corner of the screen.

This is a good time to get a feel for the size of your worksheet. The units are inches on the printed worksheet. The upper left corner is 0,0; the upper right corner is 9.50,0.00. Hence, the imaximum X distance is 9.50. The lower left corner is 0.00,7.00. Hence the maximum Y distance is 7.00. It comes as no surprise that the lower right corner is 9.50,7.00.

Grid Prameters

If you select Grid Parameters from the SET menu, you'll see three choices: Grid References, Stay on Gid and Visible Grid Dots. Selecting any of the three produces a YES/NO menu. Each toggles n environmntal setting.

Action: Making the Grid Visible. Press the left button. The hightlight is on AGAIN. Press the left button again, and the SET menu appears.

Select Grid Parameters; that is, put the hightlight on Grid Parameters and press the left button. Then, select Visible Grid Dots. Then select YES. The grid appears on the screen. The spacing between the dots represents 0.1 inch on the printed worksheet.
Return to the main command menu by pressing the left button. Select AGAIN by pressing the left button once more.

Setting Grid References shows the vertical and horizontal sheet coordinates. For reference purposes, the worksheet is divided into blocks. The horizontal direction goes left to ight from 8 to 1, and the vertical direction goes top to bottom from D to A. For example, the title block is located at A.1. Setting Grid References to YES shows the D through A and the 8 through 1 along the edge of the worksheet.

The other choice, Stay on Grid, gives you finer control over the cursor. When DRAFT is set for on grid, he crsor always points to one of the grid dots. When Stay on Grid is set to YES, you can't put the cursor in between dots. However, going off grid is dangerous and should only be used when absolutely necessary. When you are off grid it's more difficult to mak good wire connections. What may look like a connection on the screen may not be connection when interpreted by the ERCHECK and NETLIST utilities.

The final choice in the SET menu, Repeat Parameters, is best left as an advanced topic. Refer to Section 4, Commands, for more information. A typical application occurs when you need to produce series of similar labels - for example address signals labeled A0 through A7.

Action: Closing the SET menu. Press the right button.

So far you have invoked DRAFT and set the environmental parameters. You have chosen the XY display in the upper righthand corner and made the grid dots visible. Now it's time to get some parts.

8.3.3. Using the Libraries

This module uses parts from TTL.LIB. The custom library mentioned earlier is needed in Module 2.

Action: Extracting a Part from a Library. Press the left button. Move the mouse to highlight the GET command. Press the left button. The Get? prompt appears. Press the left button agin. You see a list of the configured libraries. Selct TTL.LIB: that is, put the highlight on TTL.LIB and press the left button A part list appears. Scroll down through the part list by moving the mouse until the highlight is on 138. Then, press the left button. Another menu appears. Select 74LS138.

You have extracted the part 74LS138 from the library. You can see it on your screen, but it's not yet part of your worksheet because you haven't "placed" it. The part moves with your mouse. When it moves, it appears as an outlined symbol. This is a part symbol wih much o the detail removed. If you let the mouse stay in one postion for a few seconds, the detailed part

appears, only to become the outlined symbol again if you move.

Action: Placing the Part on he Worksheet. Move the mouse until tthe XY coordinates in the upper righthand corner read 1.80,2.40. Then, press the left button. Another menu appears. The hightlight is on Place. Press the left button again and move the mouse. The part is now on the worksheet, and you have another outlined symbol. You're ready to place another copy of the part. Move the symbol until the XY coordinates read 3.00,3.90. Press teh left button twice to place another symbol on the worksheet. Now press the right button. This returns you to the main command level. Although you are at the main lvel, the main command menu does not appear until you press the left button.

When you selcted the library TTL.LIB, you saw list of the whorthand strings contained in the library. Shorthand strings are the part names without the prefixes. When you selected 138, another mnu appeared. This on listed all the valid parts, constructed by adding the valid prefixes to 138.

Instead of going through the menus, you could instead heva typed in the complete part name at the Get? prompt. For example, you could have typed 74LS138. DRAFT would then search through the configured libraries and deliver the part; DRAFT would search the libraries in the order you followed when you added them during configuration. If, at the Get? prompt you type 138, you see the menu that lists all the valid parts with 138 as a suffix.

Deleting Parts

You now have two parts on the worksheet. You can delete them or move them. To delete a part, selct the DELETE command from the main command menu. This displays the DELETE menu, which presents you with three options: Object, Block, and Undo. If you want to delete a part, choose Object and move the mouse until the cursor is on the part you want to delete.

Now you can delete the object in two ways. You can press <d> on your keyboard. Or you press the left button twice, once to display the menu and again to choose Delete If you had set the environmental characteristics left Button to YES, pressing the left button once would accomplish the same result.

Because of the way the deletion is done, som dots may remain on the screen. They are not really on the worksheet. Pressing the <ESC> key or the right button causes the screen to redraw, and th extra dots disappear.

Don't be hesitant about deleting one of these parts just for practice. You can restore the part by selecting DELETE from the main command menu and Undo from the DRLETE menu.

Moving Parts

The Move command is part of the BLOCK menu. You can move an object or define an arbitrary block on your worksheet for moving. For example, to move on of the 74LS138 decoders, perform the following steps.

.pn230

Select the BLOCK command from the main command menu; that

is, highlight BLOCK and press the left button. From the BLOCK menu, select Move. The commands now available for execution are listed across the top of the screen on the prompt line.

Put the cursor on the object you want to move. Then, press the left button to get the next menu. Select Begin. Another set of commands appear across the prompt line. Press the left button to get the menu and selct End. Now move the mouse to new location; the part follows. Press the left button twice to place the part in the new location.

If you practice moving one of the decoders, be sure to put it back.

Placing Five More Parts

You can place these parts the same way you did the 74LS138. You want to place four 74LS08s and one 74LS00.

Action: Extracting Another Part from TTL.LIB. From the main command menu, select GET. Then, select the library TTL.LIB then 08, then 74LS08.
Notice that the part appears as an AND gate. To make the schematic for this example, you want the part to appear as a NOR with negated inputs. These two representations are DeMorgan equivalents.
To get the DeMorgan equivalent, press the left button, highlight Convert, and press the left button again. Then, place the part as you would repeat the above procedure, but highlight Normal instead of Convert.

The commands along the prompt line should read as follows.

Place Rotate Convert Normal Up Over Down Mirror
Find Jump Zoom escape

If you are not at this point in the menu structure, review the last two Actions, especially the on that describes how to extract the 74LS138. What you want to do is extract a 74LD08, but convert it to its De Morgan equivalent before placing it.

Action: Placing Four More Parts. Move the gate to the XY position, 4.80,3.30. Press the left button, highlight Place, and press the left button again.
Place the next part at XY position 4.80,3.90. Plce the third at 4.80,4.50 and the fourth at 4.80,5.10. Press the right button to return to the main command level.
Although you are at the main command level, the main command menu does not appear. Press the left button. The main command menu now appears.

Module 1 requires one more part, a 7LS00 for the refresh gate. As you've probablu already figured out from looking at Figure 8-3, the first 74LS138 decodes address lines A17 and A18 into the four column address strobes, CAS\0 through CAS\1. The second decoder produces the corresponding row address strobes. The CAS\ signals are asserted during either a memory access or a refresh cycle.

Action: Placing One More Part. From the main command menu,

select GET. Enter 74LS00 at the Get? prompt. Move the cursor to the XY location 3.00,5.60. Press the left button twice. Then press the right button to return to the main command level.

The instructions in this module give exact XY postions for the parts you place. Although unlike real life, this makes the constructions of the schematic easier. If you place the parts at the designated positions, you don't have to move them around as th schematic develops.

The other options on the Place menu have to do with th appearance of the screen symbol and screen location. For example, if instead of Convert, you had chosen Mirror you would get the part's mirror image. Find the jump quick move the cursor to a different location on the screen. Choosing escape returns you to the main command menu; it's the same as pressing the right button.

Zoom gives you an overwiew of the entire sheet. It's the same ZOOM that appears as part of the main command menu. This is a good time to try it.

Action: Using ZOOM. Press the left button to display the main command menu. Select ZOOM. The ZOOM menu appears. Highlight Select, press the left button, highlight 2, and press the left button again. You can see your whole worksheet on the screen.
To return to normal size, press the left button, highlight AGAIN, and press the left button once more. The ZOOM menu reappears. Selecting AGAIN always selects the last item you selected.
You can choose Select as before, except this time, choose a 1 for normal size. Or you can select In(1). The number in parentheses indicates the level you will zoom into, and 1 is where you began.

So far in this module, you placed seven parts. One part necessitated choosing the DeMorgan equivalent.

8.3.4. Connecting Wires

If you look at Figure 8-1, you'll see that what's left to do in Module 1 is to connect the parts with wires and fill in the block. To draw wires, use the PLACE command, select Wire, and draw the wire with the mouse.

Action: Placing Wires. At the main command menu, select PLACE; that is, highlight PLACE command and press the left button. Then, select Wire. A command list appears on the prompt line.
Move the cursor until it points to the right end of the lead on pin 11 of the second 74LS138. This is XY location 4.00,4.40.
Press the left button. Select Begin; that is, highlight Begin and press the left button. Move the mouse and draw a wire to XY location 4.30,4.40.
Press the left button twice - you just selected Begin again to change direction. Draw a wire straight up to XY location 4.30,3.40. Then select Begin again and bring the wire to pin 1 on the top 74LS08.

Press the left button and select New. You're ready to draw another wire. Place the cursor at the appropriate position. Daw the remaining wires shown in Figure 8-1.

General Procedure for Drawing Wires. The procedure assumes that you selected PLACE from the main command menu and then selected Wire. The command list along the prompt line reads as follows: Begin End Find Zoom Escape.

1. Press the left button twice to start drawing.
2. When you want to change direction, press the left button twice.
3. When you want to start a new wire, press the left button, highlight. New, and press the left button again. A new wire is one that's not connected to an existing wire. Selecting New allows you to place the start of another wire at a different location. It's like picking up the pen when drawing.
4. When you've finished drawing the last wire, press the left button, highlight End and press the left button again. This returns you to the main command level.

A Macro for Placing Wires. Placing wires is a task you will perform frequently when drawing schematics. Although Module 2 describes how to make a macro, it makes sense to include here a short description of a macro that makes placing wires easier. When this macro, one key gets you from the main command level to the point where moving the cursor draws the wire.

To define the macro on function key <F2>, perform the following steps.

1. From the main command level, select MACRO. Then, select Capture. The message <macro> appears on the prompt line.
2. Press function key <F2>. Then press <ENTER>.
3. Press the keys <P>, <W> and <B>, for PLACE, Wire and Begin.
4. End the macro capture by pressing the <CTRL> and <END> keys simultaneously. The message <<<MACRO END>>> appears on the prompt line.

After defining a macro, you can use it during the current DRAFT sesson. To save it for use in future DRAFT sessions, you must write the macro out to a file. To do that, select MACRO from the main command menu; then, select Write. The prompt line diplays the message. Write all macros to? Respond with the name of the file to which you want to write the macro.

To use the macro, first place the cursor where you want to start drawing the wire. Press <F2>. When you want to change direction, press <F2> again. When you've finished drawing a wire, press the left button and select End.

N O T E

When connecting wires end-to-end, take care not to overlap them. Overlapping wires do not form a connection. The CLEANUP utility will repair overlapping wires and make them into connections.

However, CLEANUP will not fix a wire that overlaps with the pin on a body object. For example, when you connected the wire to pin 11 on the 74LS138, you connected a wire to a body object pin.

Action:

Placing More Wires. Look at Figure 8-1 and place the remaining wires.

Place a wire from pin 10 of the lower 74LS138 to pin 1 of the second 74LS08. The wire is vertical along X = 4.40.

Place a wire from pin 8 of the lower 74LS138 straight across to pin 1 of the third 74LS08.

Place a wire from pin 7 of the lower 74LS138 to pin 1 of the fourth 74LS08. The wire is vertical along X = 4.30.

Place a wire from pin 2 of the first 74LS08 all the way down to (4.5,5.8). Then turn left and connect the wire to pin 3 on the 74LS00.

Place a wire from pin 1 of the 74LS00 to (1.00,5.70).

Place a wire from 6 of the lower 74LS138 to (1.00,4.50). Then place another wire from pin 5 to (1.30,4.70). Then connect pin 4 to pin 1 of the 74LS00. This wire is vertical along X = 2.50. This is not a true connection until you place a junction. Placing a junction comes later.

Place a wire from pin 2 of the 74LS00 to pin 6 of the lower 74LS138. This wire is vertical along X = 2.3.

Place a wire from pin 4 of the upper 74LS138 to pin 5 of the lower 74LS138. This wire is vertical along X = 1.30. Again, this is not a true connection until you place a junction.

Place wires from pins 1, 2, 6 and 5 of the upper 74LS128 to X = 1.0.

.PN234

7.6.2. SAMPLE 2.SRC: A TTL Block Symbol Library Source File

{begining of the source file, SAMPLE 3.SRC}

```
PREFIX
'74LS'    -    'LS'
'74S'     -    'S'
'74ALS'   -    'ALS'
'74AS'    -    'AS'
'74HCT'   -    'HCT'
'74HC'    -    'HC'
'74ACT'   -    'ACT'
'74AC'    -    'AC'
'74F'     -    'F'
'74'
END

'7442' '74LS42' '74HC42' '7443' '7444'
5         11        1
L4        1         IN             'A'
L5        15        IN             'B'
L6        13        IN             'C'
L7        12        IN             'D'
R10       11        DOT OUT        '9'
R9        10        DOT OUT        '8'
R8        9         DOT OUT        '7'
R7        7         DOT OUT        '6'
R6        6         DOT OUT        '5'
R5        5         DOT OUT        '4'
```

```
R4        4         DOT OUT        '3'
R3        3         DOT OUT        '2'
R2        2         DOT OUT        '1'
R1        1         DOT OUT        '0'
T0        16        PWR            'VCC'
B0        8         PWR            'GND'

'7446' '7447' '74LS47'
6         8         1
L1        4         IN             'R\B\0\'
L2        5         IN             'R\B\1\'
L3        3         IN             'L\T\'
L4        7         IN             '1'
L5        1         IN             '2'
L6        2         IN             '4'
L7        7         IN             '8'
R7        14        DOT OC         'G'
R6        15        DOT OC         'F'
R5        9         DOT OC         'E'
R4        10        DOT OC         'D'
R3        11        DOT OC         'C'
R2        12        DOT OC         'B'
R1        13        DOT OC         'A'
T0        16        PWR            'VCC'
B0        8         PWR            'GND'

'7474'   '74ALS74'  '74AS74' '74LS74' 74S74'
'74HC74' '74AC74'
6         6         2
L2        2         12             IN        'D'
L4        3         11             CLK IN    'CK'
B3        1         13             DOT IN    'CL'
T3        4         10             DOT IN    'P'
R4        6         8              OUT       'Q\'
R2        5         9              OUT       'Q'
T0        14        14             PWR       'VCC'
B0        7         7              PWR       'GND'
```

{This part has two parts per package. The far left column represents the pin position. The second column contains the pin number of the first D flip-flop in the packagem The third column contains the pin number of the second D flip-flop in the package. Note tthe power pins on 14 and 7. They are required for both devices.}

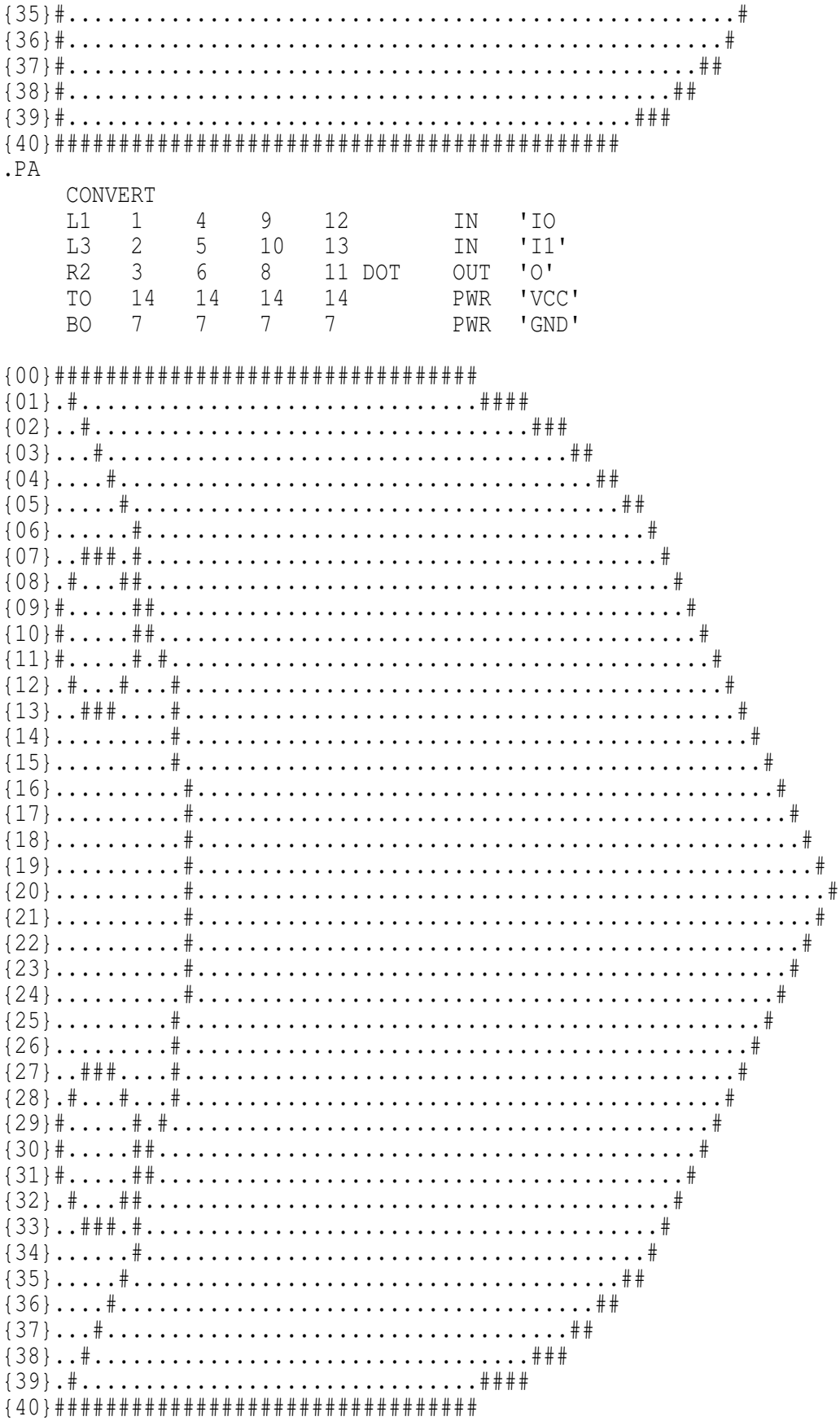
```
'74ALS160' '74AS160' '74LS160' '74HC160'
'74ALS161' '74AS161' '74LS161' '74HC161'
'74ALS162' '74AS162' '74LS162' '74S162' '74HC162'
'74ALS163' '74AS163' '74LS163' '74S163' '74HC163'
8         10        1
B4        1         DOT IN    'CL'
L1        9         DOT IN    'LD'
L2        10        IN        'ENT'
L3        7         IN        'ENP'
L4        2         CLK IN    'CLK'
L6        3         IN        'A'
L7        4         IN        'B'
L8        5         IN        'C'
L9        6         IN        'D'
R9        11        OUT       'QD'
```

```
     R8        12        OUT       'QC'
     R7        13        OUT       'QB'
     R6        14        OUT       'QA'
     R4        15        OUT       'C0'
     T0        16        PWR       'VCC'
     B0        8         PWR       'GND'

     {End of the source file, SAMPLE 2.SRC}
```

7.6.3.SAMPLE 3.SRC: A Bitmap Symbol Library Source File

(Beginning of the source file, SAMPLE 3.SRC)

```
.pa
     PREFIX
     END

     '7400'
     6    4    4
     L1   1    4    9    12        IN   'IO
     L3   2    5    10   13        IN   'I1'
     R2   3    6    8    11 DOT    OUT  'O'
     TO   14   14   14   14        PWR  'VCC'
     BO   7    7    7    7         PWR  'GND'

{00}##############################################
{01}#.............................................###
{02}#...............................................##
{03}#................................................##
{04}#..................................................#
{05}#...................................................#
{06}#....................................................#
{07}#.....................................................#
{08}#......................................................#
{09}#.......................................................#
{10}#.......................................................#
{11}#........................................................#
{12}#........................................................#
{13}#.........................................................#
{14}#.........................................................#
{15}#.........................................................#
{16}#..........................................................#
{17}#.........................................................#
{18}#.........................................................#
{19}#.........................................................#
{20}#.........................................................#
{21}#.........................................................#
{22}#.........................................................#
{23}#.........................................................#
{24}#.........................................................#
{25}#.........................................................#
{26}#.........................................................#
{27}#.........................................................#
{28}#.........................................................#
{29}#.........................................................#
{30}#.......................................................#
{31}#.......................................................#
{32}#.....................................................#
{33}#....................................................#
{34}#...................................................#
```



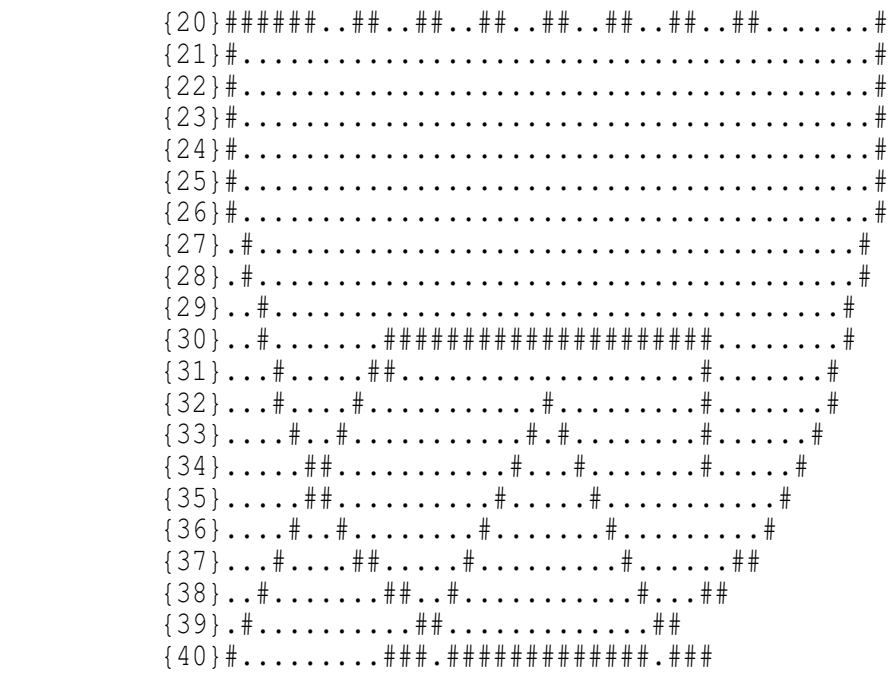
```
{35}#................................................#
{36}#...............................................#
{37}#..............................................##
{38}#............................................##
{39}#.........................................###
{40}##########################################
.PA
     CONVERT
     L1   1    4    9    12        IN   'IO
     L3   2    5    10   13        IN   'I1'
     R2   3    6    8    11 DOT    OUT  'O'
     TO   14   14   14   14        PWR  'VCC'
     BO   7    7    7    7         PWR  'GND'

{00}#################################
{01}.#............................####
{02}..#.........................###
{03}...#.....................##
{04}....#..................##
{05}.....#...............##
{06}......#............#
{07}..###.#.........#
{08}.#...##.....#
{09}#.....##.....#
{10}#.....##.....#
{11}#.....#.#.....#
{12}.#...#...#.....#
{13}..###....#.....#
{14}.........#.....#
{15}.........#.....#
{16}.........#.....#
{17}.........#.....#
{18}.........#.....#
{19}.........#.....#
{20}.........#.....#
{21}.........#.....#
{22}.........#.....#
{23}.........#.....#
{24}.........#.....#
{25}.........#.....#
{26}.........#.....#
{27}..###....#.....#
{28}.#...#...#.....#
{29}#.....#.#.....#
{30}#.....##.....#
{31}#.....##.....#
{32}.#...##.....#
{33}..###.#.....#
{34}......#.....#
{35}.....#.....##
{36}....#.....##
{37}...#.....##
{38}..#.....###
{39}.#.....####
{40}#################################

     '74ALS10'
     6    4    3
     L1   1    3    9              IN   'IO
     L2   2    4    10             IN   'I1'
```

```
     L3   13   5    11             IN   'I2'
     R2   3    6    8   DOT       OUT  'O'
     TO   14   14   14             PWR  'VCC'
     BO   7    7    7              PWR  'GND'
     BITMAP '7400'
     CONVERT
     L1   1    3    9              IN   'IO
     L2   2    4    10             IN   'I1'
     L3   13   5    11             IN   'I2'
     R2   3    6    8              OUT  'O'
     TO   14   14   14             PWR  'VCC'
     BO   7    7    7              PWR  'GND'

{00}################################
{01}.#.............................####
{02}..#.................................###
{03}...#.....................................##
{04}....#.....................................##
{05}.....#.....................................##
{06}......#.....................................#
{07}..###.#.....................................#
{08}.#...##.....................................#
{09}#.....##....................................#
{10}#.....##....................................#
{11}#.....#.#...................................#
{12}.#...#...#..................................#
{13}..###....#..................................#
{14}.........#..................................#
{15}.........#..................................#
{16}..........#.................................#
{17}.....###..#.................................#
{18}....#...#.#.................................#
{19}...#.....##.................................#
{20}####.....##.................................#
{21}...#.....##.................................#
{22}....#...#.#.................................#
{23}.....###..#.................................#
{24}..........#.................................#
{25}.........#..................................#
{26}.........#..................................#
{27}..###....#..................................#
{28}.#...#...#..................................#
{29}#.....#.#...................................#
{30}#.....##....................................#
{31}#.....##....................................#
{32}.#...##.....................................#
{33}..###.#.....................................#
{34}......#.....................................#
{35}.....#.....................................##
{36}....#.....................................##
{37}...#.....................................##
{38}..#.....................................###
{39}.#.............................####
{40}################################
.PA
     '7402'
     6    4    4
     L1   2    5    8    11        IN   'IO
     L3   3    6    9    12        IN   'I1'
     R2   1    4    10   13 DOT    OUT  'O'
```

```
     TO   14   14   14   14        PWR  'VCC'
     BO   7    7    7    7         PWR  'GND'

{00}##################################
{01}.#..............................####
{02}..#.................................###
{03}...#.....................................##
{04}....#....................................##
{05}.....#.....................................##
{06}......#.....................................#
{07}......#.....................................#
{08}.......#....................................#
{09}.......#.....................................#
{10}#########....................................#
{11}........#....................................#
{12}.........#....................................#
{13}.........#....................................#
{14}.........#....................................#
{15}.........#....................................#
{16}..........#...................................#
{17}..........#...................................#
{18}..........#...................................#
{19}..........#...................................#
{20}..........#...................................#
{21}..........#...................................#
{22}..........#...................................#
{23}..........#...................................#
{24}..........#...................................#
{25}.........#....................................#
{26}.........#....................................#
{27}.........#....................................#
{28}.........#....................................#
{29}........#.....................................#
{30}#########....................................#
{31}.......#.....................................#
{32}.......#.....................................#
{33}......#......................................#
{34}......#......................................#
{35}.....#.....................................##
{36}....#.....................................##
{37}...#.....................................##
{38}..#.................................###
{39}.#..............................####
{40}##################################
.PA
     CONVERT
     L1        2         5         8    11   DOT       IN'IO'
     L3        3         6         9    12   DOT       IN'I1'
     R2        1         4         10   13             OUT'O'
     T0        14        14        14   14   PWR'VCC'
     B0        7         7         7    7    PWT'GND'
     BITMAP '7400'

     '7433' 74LS33' '74ALS33'
     6         4         4
     L1        2         5         8    11             IN'IO'
     L3        3         6         9    12             IN'I1'
     R2        1         4         10   13   DOT       OUT'O'
     T0        14        14        14   14   PWR'VCC'
     B0        7         7         7    7    PWT'GND'
```

```
     BITMAP '7402'

     CONVERT
     L1        2         5         8    11   DOT       IN'IO'
     L3        3         6         9    12   DOT       IN'I1'
     R2        1         4         10   13             OUT'O'
     T0        14        14        14   14   PWR'VCC'
     B0        7         7         7    7    PWT'GND'
     BITMAP '7400'

{The following are electrical and electronic parts}

'ANTENNA'
2              1         0
B1             OUT       'ANT'
{ 0}####################
{ 1}.#........#........#.
{ 2}..#.......#.......#..
{ 3}...#......#......#...
{ 4}....#.....#.....#....
{ 5}.....#....#....#.....
{ 6}......#...#...#......
{ 7}.......#..#..#.......
{ 8}........#.#.#........
{ 9}.........###.........
{10}..........#..........

'CAP'
'CAPACITOR'
REFERENCE 'C'
{X Size =} 2    {Y Size =} 1  {Parts per Package =} 0
T1        SHORT    PAS   '1'
B1        SHORT    PAS   '2'
{ 0}..........#..........
{ 1}..........#..........
{ 2}.####################
{ 3}......................
{ 4}......................
{ 5}......................
{ 6}......................
{ 7}......#########......
{ 8}...###....#....###...
{ 9}.##.......#.......##.
{10}..........#..........


'GND POWER'
{X Size =} 2    {Y Size =} 1  {Parts per Package =} 0
T1        PWR            'GND'
{ 0 }####################
{ 1}......................
{ 2}......................
{ 3}...###############...
{ 4}......................
{ 5}......................
{ 6}.....##########......
{ 7}......................
{ 8}......................
{ 9}.........###.........
{10}......................
```

```
'MOSFET N'
REFERENCE 'Q'
3         1         0
L2        SHORT     IN        'GATEr
T2        SHORT     IN        'DRAIN'
B2        SHORT     IN        'SOURCE'
{ 0}.........#..........#..........
{ 1}.........#..........#..........
{ 2}....#....############..........
{ 3}....#....#.....................
{ 4}....#....#.....................
{ 5}....#....#.....#...............
{ 6}....#....#....##...............
{ 7}....#....#...###...............
{ 8}....#....#..####...............
{ 9}....#....#.#####...............
{10}....#....############..........
{11}....#....#.#####....#..........
{12}....#....#..####....#..........
{13}....#....#...###....#..........
{14}....#....#....##....#..........
{15}....#....#.....#....#..........
{16}....#....#..........#..........
{17}....#....#..........#..........
{18}....#....############..........
{19}....#....#..........#..........
{20}#####....#..........#..........

'TRIODE'
4         4         0
L2        IN        'GRID'
B0        IN        'CATHODE'
B1        IN        'FILAMENT 1'
B3        IN        'FILAMENT 2'
T2        IN        'PLATE'

.PA

{00}..............#############
{01}............##.............##
{02}..........##.................##
{03}........##.....................##
{04}.......#.........................#
{05}......#...........................#
{06}.....#.............................#
{07}....#...............................#
{08}...#.................................#
{09}...#.................................#
{10}..#...................................#
{11}..#...................................#
{12}.#.....................................#
{13}.#.....................................#
{14}#.......................................#
{15}#.......................................#
{16}#.......................................#
{17}#.......................................#
{18}#.......................................#
{19}#.......................................#
```



.PA
.PN244

8.5.2 BACKANNO

Assume that, in Figure 8-4, you really wanted U2B to be designated as U2A. You also wanted U2C to be U2A to be U2C. You may find that your board lays out better with a pin rearrangement on the 74LS00 designated as U2.

You can construct a was/is file and use the BACKANNO utility. A was/is file is a text file that lists the old reference designator followed by what you want the designator to be. For example, here is a was/is file that redefines the U2 74LS00.

```
U2B       U2A
U2C       U2B
U2A       U2C
```

When BACKANNO changes the reference designators it updates the pin numbers correspondingly. To run BACKANNO on the file tutor.sch, type the following.

```
C:\ORCAD>BACKANNO TUTOR.SCH WASIS/O
```

This example assumes that tutor.sch is in the sheet directory and that the wasis file is in the DRAFT directory.

8.5.3 CLEANUP

Typically, CLEANUP is the second utility you run. CLEANUP corrects simple drawing errors that may inadvertently show up as errors in ERCHECK's electrical rules check. Note that CLEANUP will not fix up wires that overlap body object pins. To run CLEANUP on the work file tutor.sch, type the following.

```
C:\ORCAD>CLEANUP TUTOR.SCH/O
```

This example assumes that tutor.sch is in the sheet directory.

8.5.4 ERCHECK

This utility performs a basic electrical rules test. To run ERCHECK on the work file tutor.sch, type the following.

```
C:/ORCAD>ERCHECK TUTOR.SCH TUTOR.ERR/O
```

This example assumes that tutor.sch is in the sheet directory. The second filename is for a file that will contain messages from ERCHECK. It is created in the DRAFT directory. If you omit the second filename, these messages display on the console. ERCHECK lets you know that it is working by displaying a sequence of asterisks (*) and periods (.).

A typical test is to ensure that all inputs to a device are connected. For example, if you negleted to connect pin 3 of the upper decoder to ground, you get the following message.

```
WARNING - INPUT has NO Driving Source U3,C
```

Also, you must label every bus. For example, if in tutor.sch, yo, did not include the bus label CAS\[0..3], ERCHECK would display the following message.

```
<<<WARNING>>>Unconnected MODULE PORT "CAS\[0..3]"
at X = 8.10 at Y = 2.90
```

By connecting the two powerobjects, VCC and + 5V, together, you generate the following waming.

```
WARNING - POWER Suplies are CONNECTED VCC <-> + 5V
```

This is something you intended. You can safely ignore the warning.

8.5.5 NETLIST

The NETLIST utility generates a netlist in a variety of formats, the default of which is EDIF.

To create a proper net list, you need to show care when dealing with bus labels, module ports, and power objects. Refer to Section 6 for a detailed explanation of these requirements. Here are some general guidelines.

* All buses must be labeled. The signals making up the bus must have bus member labels. For example, in this tutorial, CAS\[0..3] is the bus label and CAS\0 through CAS\3 are the bus member labels. the bus module port name need not be the same as the bus label.

* Signals that go off-sheet must do so via module ports.

Note that earlier releases of OrCAD/SDT allowed the use of "public labels".

* Most devices in the parts libraries have hidden power and ground pins. To make a netlist connection to these pins, the worksheet must contain a power object. For example, many devices have a VCC pin that does not appear in its symbolic representation. For NETLIST to make a connection to VCC, a power object whose value is VCC must appear on the worksheet.

To run NETLIST on the work file tutor.sch and generate a file in CALAY format, type the following.

```
C:\ORCAD>NETLIST TUTOR.SCH TUTOR.NET CALAY /S /O
```

This example assumes that tutor.sch is in the sheet directory. The second filename is for a file that will contain the net list. This file is created in the DRAFT d7rectory. If you omit the second filename, the net list displays on the console.

The /s signifies calay as a special format. Note that when you choose CALAY format, NETLIST creates two files. One file contains the CALAY netlist; this is the file tutor.net specified on the command line. The other file is the component file. Its name is the name of the original file with the CMP extension.two files.

NETLIST lets you know that it is working by displaying a sequence of asterisks (*) and periods (.). The file containing the CALAY netlist is as follows.

```
/N00001 U2(6)  U4(6) U5(1) U2(2);
/N00002 U5(12) U2(10);
/N00003 U5(8) U2(9);
/N00004 U2(8) U3(5);
/N00005 U1(1) U4(11);
/N00006 U1(2) U1(5) U1(10) U1(13) U2(3);
/N00007 U1(4) U4(10);
/N00008 U4(9) U1(9);
/N00009 U4(7) U1(12);
/N00010 U2(4);
/N00011 U2(5);
/NOOO12 U5(10);
/N00013 U4(1) U3(1);
/NOOO14 U4(2) U3(2);
/GND    U3(3) U2(7) U5(7) U3(8) U1(7) U4(8);
/N00016 U3(6);
/N00017 U3(4) U4(5);
/N00018 U1(3);
/N00019 U1(6);
/VCC    U4(3) U2(14) U5(14) U3(16) U1(14) U4(16);
/N00021 U1(8);
/N00022 U4(4) U2(1);
/N00023 U1(11);
/N00024 U3(15);
/N00025 U3(14);
/N00026 U3(13);
/N00027 U3(12);
```

The component file net.cmp is as follows.

```
74LS00          U2          shape      -X-        -Y-  0
74LS08          U1          shape      -X-        -Y-  0
74LS138         U3          shape      -X-        -Y-  0
TIME DELAY      U5          shape      -X-        -Y-  0
```

8.5.6 PARTLIST

This utility creates a list of all the parts used in the worksheet. You can specify an Include file if you want that list to contain additional information. To run PARTLIST on tutor.sch iwithout an Include file, type the following.

```
C:\ORCAD.PARTLIST TUTOR.SCH TUTOR.1ST/O
```

This example assumes that tutor.sch is in the sheet directory. The second filename is the name of a file that will contain the part list. This file is created in the DRAFT directory. If you leave out the second filename, the part list displays on the console. The part list that results from tutor.sch follows.

```
 Memory Decode, Module 2 of Tutorial  Revised: November 16, 1986
 11111                                Revision:
 Bi11 Of Materials       November 16, 1986   22:18:08    Page  1

     Item      Quantity       Reference      Part
    ------------------------------------------------
       1            1         U1             74LS08
       2            1         U2             74LS00
       3            2         U3,U4          74LS138
       4            1         U5             TIME DELAY
```

An Include file is a text file that contains information you want to add to the part list. It consists of a series of columns with a header.

The header begins with a pair of single quotes with no character between them. The remainder of the header consists of the titles for the additional columns. These titles will be added to the header information in the part list.

The rows underneath the header consist of part-specific information. The first row identifies the part, and the subsequent rows depend on the associated header. An Include file for tutor.sch follows.

```
''                      DESCRIPTION              Part Order Code
'74LS00'        TTL Quad Two Input NAND Gate     10003040000
'74LS08'        TTL Quad Two Input AND Gate      10003050000
'74LS138'       TTL Decoder/Demultiplexer        10005120000
'TIME DELAY 'Gate Delay 25/125                   10006400000
```

To specify this Include file when you invoke PARTLIST type

the following.

```
C:\ORCAD>PARTLIST TUTOR.SCH TUTOR.1ST
TUT.INC /I /O
```

Note that tutor.sch is in the sheet directory (C:\orcad\sheet) and that tutor.lst and tut.inc are in the DRAFT directory (C:\orcad).

The resulting part list now looks as follows.

```
 Memory Decode, Module 2 of Tutorial  Revised: November 16, 1986
 11111                                  Revision:
 Bill Of Materials      November 16,  1986       22:18:28 Page 1

Item Qty Ref    Part      DESCRIPTION           Part Order Code
-------------------------------------------------------------
1    1   U2,    74LS00   TTL Quad Two Input NAND Gate 10003040000
2    1   U1     74LS08   TTL Quad Two Input AND Gate  10003050000
3    2   U3,U4  74LS138  TTL Decoder/Demultiplexer    10003060000
4    1   U5     TIME DELAY Gate Delay 25/125          10006400000
```

8.5.7 PRINTALL, PLOTALL

These utilities permit the printing or plotting of files in batch mode.

With the PRINTALL utility, you can print the schematic file without invoking DRAFT. For example, to print the file tutor.sch, type the following.

```
C:\ORCAD>PRINTALL TUTOR.SCH /O /S
```

The /s switch selects scale mode. By default, the print mode is compressed.

The PRINTALL utility is most useful for hierarchical structures and that file structures. For example, tutor.sch is a one-sheet schematic that could be member of a flat file structure. A flat file structure is a text file containing the names of a number of one-sheet schematics.You can print the entire collection by invoking PRINTALL and specifying the flat file structure.

The PLOTALL utilly is similar to PRINTALL, but is specified to plotters. Note that DRAFT's Hardcopy command works with printers, not plotters. To plot a schematic file, refer to the description of the PLOTALL utility in Section 6.

8.6 Conclusion

This is the end of the tutorial for OrCAD/SDT. As an introductory tutorial, it was not intended as a complete description of the OrCAD/SDT environment. The tutorial covered the basics of creating a one-sheet schematic and using the OrCAD utilities. The OrCAD/SDT environment is rich and you will discover many additional features as you build your design.

For example, the tutorial has only scratchet the surface of macro building and has hardly mentioned hierarchical structures. The Drag command provides additional editing capability. With the

Fixup command, you can make the wires rubberbanded with Drag orthogonal again.

Examples of other features include the ability to improve cursor movement by setting tags on your worksheet and to speed up the repetitive placement of objects with the REPEAT command.

Consult the other sections in this manual for additional information.
.PN250
.HEAppendix A Schematic Design ToolsPD

APPENDIX A

SUMMARY OF LIBRARY COMPONENTS

Directory of ANALOG.LIB

1458	1558	7621	ADC0801
ADC0802	ADC0803	ADC0804	ADC0805
ADC0808	ADC0809	DAC0800	DAC0801
DAC0802	DAC0806	DAC0807	DAC0808
DAC8408	LF13741H	LF13741N	LF147D
LF147N	LF155H	LF156H	LF157H
LF255H	LF256H	LF257H	LF347D
LF347N	LF351H	LF351N	LF353H
LF353N	LF355H	LF355N	LF356H
LF356N	LF357H	LF357N	LF411H
LF411N	LF412H	LF412N	LF441H
LF441N	LF442H	LF442N	LF444D
LF444N	LH0002N	LH0003H	LH0004H
LM101H	LM101J	LM101J14	LM102H
LM107H	LM107J	LM107J14	LM108H
LM108J	LM108J8	LM10H	LM10N
LM11D	LM11H	LM11N	LM124
LM13080N	LM13080P	LM1458H	LM1458J
LM1458N	LM146J	LM148J	LM149
LM1558H	LM1558J	LM158H	LM159J
LM192H	LM192J	LM201H	LM201J
LM201J14	LM202H	LM207H	LM207J
LM207J14	LM208H	LM208J	LM208J8
LM216H	LM224	LM246J	LM246N
LM248	LM248J	LM258H	LM2900J
LM2900N	LM2904N	LM2924J	LM292N
LM292H	LM292J	LM301H	LM301J
LM301J14	LM301N	LM302H	LM307H
LM307J	LM307J14	LM307N	LM308H
LM308J	LM308J8	LM316H	LM324
LM3301N	LM339	LM3401N	LM346J
LM346N	LM348	LM348J	LM348N
LM349N	LM358H	LM358N	LM359J
LM359N	LM3900N	LM392H	LM392J
LM392N	LM4250H	LM4250J	LM555
LM709H	LM709N	LM709N8	LM725H
LM725N	LM741	LM741H	LM741J14
LM741N	LM741N14	LM747	LM747H
LM747J	LM748H	LM748J	LM748N
LM7805	LM7806	LM7808	LM7810
LM7812	LM7815	LM7818	LM7824
LM7885	MC1455	MC1555	MC1741

NE555	OPAMP	TL072	TL082
TL092	UA741	VOLTAGE REGULATOR	

Directory of ASSEMBLY.LIB

08 PIN	14 PIN	16 PIN	24 PIN
28 PIN	40 PIN BOTTOM	40 PIN TOP	CAP 2X7
EDGE CONNECTOR	RESISTOR 1X2	RESISTOR 1X3	RESISTOR 1X5
RESISTOR PACK	TO 220	TO 39	TO 92

Directory of CMOS.LIB

4000	4001	4002	4006	4007	4008	4009	4010	4011
4012	4013	4014	4015	4016	4017	4018	4019	4020
4021	4022	4023	4024	4025	4026	4027	4028	4029
4030	4031	4032	4033	4034	4035	4037	4038	4040
4041	4041	4043	4044	4045	4046	4047	4048	4049
4050	4051	4052	4053	4054	4055	4056	4057	4059
4060	4063	4066	4068	4069	4070	4071	4072	4073
4075	4076	4077	4078	4081	4082	4085	4086	4089
4093	4094	4095	4096	4098	4099	4501	4502	4503
4504	4505	4506	4508	4510	4511	4512	4513	4514
4515	4516	4517	4518	4519	4520	4521	4522	4524
4526	4527	4528	4529	4530	4531	4532	4534	4536
4537	4538	4539	4541	4543	4544	4547	4549	4551
4552	4553	4554	4555	4556	4557	4558	4559	4560
4561	4562	4566	4568	4569	4572	4573	4574	4575
4580	4581	4582	4583	4584	4585	4597	4598	4599
5101	14000	14001	14002	14006	14007	14008	14011	14012
14013	14014	14015	14016	14016	14017	14018	14020	14021
14023	14024	14025	14027	14028	14029	14032	14034	14035
14038	14040	14042	14043	14044	14046	14051	14052	14053
14066	14068	14069	14070	14071	14072	14073	14075	14076
14077	14078	14081	14082	14093	14094	14099	14160	14161
14162	14163	14174	14175	14194	14501	14502	14503	14504
14505	14506	14508	14510	14511	14512	14513	14514	14515
14516	14517	14518	14519	14520	14521	14522	14524	14526
14527	14528	14529	14530	14531	14532	14534	14536	14536
14538	14539	14541	14543	14544	14547	14549	14551	14552
14553	14554	14555	14556	14557	14558	14559	14560	14561
14562	14566	14568	14569	14572	14573	14574	14575	14580
14581	14582	14583	14584	14585	14597	14598	14599	40100
40101	40102	40103	40104	40105	40106	40225	40116	40160
40161	40162	40163	40174	40175	40182	40192	40193	40194
42100	45100	45101	45104	45106	45107	45109	45112	
145104	145106	145107	145109	145112				

Directory of DEVICE.LIB

12 HEADER	1RSW10	1RSW12
1RSW3	1RSW6	2RSW12
2RSW3	2RSW5	4 HEADER
ANTENNA	BATTERY	BNC
BRIDGE	BUZZER	CAP
CAPACITOR	CAPACITOR FEED	CAPACITOR POL
CAPACITOR VAR	CIRCUIT BREAKER	COAX
CONNECTOR COAX	CONNECTOR COAX-F	CONNECTOR COAX-M
CONNECTOR DB15	CONNECTOR DB25	CONNECTOR DB9

CRYSTAL	DIODE	DIODE BREAKDOWN
DIODE SCHOTTKI	DIODE TUNNEL	DIODE VARACTOR
DIODE ZENER	DIODE ZENER1	FUSE
GND EARTH	GND FIELD SIGNAL	GND POWER
GND SIGNAL	GTO	INDUCTOR
INDUCTOR IRON	INDUCTOR IRON1	INDUCTOR ISOLATED
INDUCTOR VAR	INDUCTOR VARIABLE IRON	JEET N
JEET P	JUMPER	LAMP
LAMP NEON	LED	METER AMP
METER MA	METER MV	METER UA
METER UV	METER VOLT	METER VU
MICROPHONE	MOSFET DUAL G/N	MOSFET DUAL G/P
MOSFET N	MOSFET P	MOTOR SERVO
MOTOR STEPPER	NPN	NPN DAR
NPN DIAC	OPTO ISOLATOR	OPTO ISOLATOR-A
PHONEJACK	PHONEJACK STEREO	PHONEJACK STEREO SW
PHONEPLUG	PHOTO NPN	PHOTODIODE
PLUG AC FEMALE	PLUG AC MALE	PNP
PNP DAR	PNP DIAC	POT
R	R-PACK	RCA JACK
RELAY DPST	RELAY SPDT	RELAY SPST
RESISTOR	RESISTOR 8PACK	RESISTOR BRIDGE
RESISTOR TAPPED	RESISTOR VAR	RESISTOR VAR 2
SCR	SIGNAL AC	SOURCE CURRENT
SOURCE VOLTAGE	SPARK GAP	SPARK GAP CAP
SPEAKER	SW DIP-4	SW DIP-8
SW PUSHBUTTON	SW SPDT	SW SPST
THERMAL FUSE	THERMISTOR	TRANSFORMER
TRANSFORMER AIR CORE	TRANSFORMER CT	TRANSFORMER ISOLATED
TRANSFORMER STEPUP	TRANSFORMER VAR	TRANZORB
TRIAC	TRIAC DRIVER	TRIGGERED SKARK GAP
TRIODE	TUBE PHOTO MULTIPLIER	UJT N
UJT P	VARISTOR	

Directory of ECL.LIB

MC10100	MC10101	MC10102	MC10103	MC10104	MC10105	MC10106
MC10107	MC10109	MC10110	MC10111	MC10113	MC10114	MC10115
MC10116	MC10117	MC10118	MC10119	MC10121	MC10123	MC10124
MC10125	MC10128	MC10129	MC10130	MC10131	MC10132	MC10133
MC10134	MC10135	MC10136	MC10137	MC10138	MC10139	MC10141
F10145A	MC10153	MC10154	MC10158	MC10159	MC10160	MC10161
MC10162	MC10163	MC10164	MC10165	MC10166	MC10168	MC10170
MC10171	MC10172	MC10173	MC10174	MC10175	MC10176	MC10177
MC10178	MC10179	MC10180	MC10181	MC10182	MC10186	MC10188
MC10189	MC10190	MC10191	MC10193	MC10195	MC10197	MC10210
MC10211	MC10212	MC10216	MC10231	F10402	F10414	F10415
F10416	F10422	F10470	F10474	F10500	F10501	F10502
F10503	MC10504	MC10505	MC10506	MC10509	MC10514	MC10515
MC10516	MC10517	MC10518	MC10519	MC10521	MC10530	MC10531
MC10532	MC10534	MC10535	MC10536	MC10537	MC10538	MC10539
MC10541	MC10558	MC10559	MC10560	MC10561	MC10562	MC10563
MC10564	MC10565	MC10566	MC10568	MC10570	MC10571	MC10572
MC10574	MC10575	MC10576	MC10578	MC10579	MC10580	MC10581
MC10582	MC10586	MC10588	MC10590	MC10591	MC10593	MC10595

MC10597 MC10610 MC10611 MC10612 MC10616 MC10631 F100101
F100102 F100107 F100112 F100113 F100114 F100117 F100118
F10012 F100123 F100124 F100125 F100126 F100130 F100131
F100136 F100141 F100142 F100145 F100150 F100151 F100155
F100156 F100158 F100160 F100163 F100164 F100165 F100166
F100171 F100179 F100180 F100181 F100182 F100183 F100194
F100402 F100414 F100415 F100416 F100422 F100470 F100474
F100170/4
F100170/8

Directory of INTEL.LIB

Z80 Z80PIO 8031 80C31 8032 8035 80C35
8039 80C39 8040 8041 8042 8044 8048
80C48 8049 80C49 8050 8051 80C51 8052
8085 8086MAX 8086MIN 8087 8088MAX 8088MIN 8089
8096 8097 8155 8156 8185 8203 8205
8206 8207 8208 8212 8216 8226 8231
8237 8237A 8243 8251 8251A 8253 8254
82C54 8255 82C55 8256 8257 8259A 8272
8272A 8273 8274 8275 8276 8279 8282
8283 8284 8286 8287 8288 8289 8291
8292 8294 8294A 8295 8344 8396 8397
8641 8741 8742 8744 8748 8749 8751
8755 80186 80188 80286 80287 80386 8206-2
82062 82064 82188 82284 82288 82289 82384
82501 82530 82586MA 82586MI 82588HI 82588HM 82720
82731

Directory of MEMORY.LIB

10H8 10L8 12H6 12L6 14H4 14L4 14L8 16C1
16H2 16L2 16A4 16R4 16RP4 16X4 16L6 16PR6
16R6 16H8 16HD8 16L8 16LD8 16P8 16R8 16RP8
16V8 16Z8 18L4 18P8 20C1 20L2 20R4 20X4
20R6 20L8 20R8 20V8 20X8 315 406 426
12L10 1400 1420 1421 1430 1600 1601 18S42
18S46 18SA46 20L10 20X10 2015 2016 2018 2019
2063 2064 2068 2069 2114 2115 2125 2130
2141 2147 2148 2149 2164 2167 2168 2186
2187 22V10 24S10 24SA10 24S41 24SA41 24S81 2600
2620 2630 27S12 2716 27S18 27S19 27S20 27S21
27S43 2764 27C64 2816 2817 2864 28S86 3628
3636 3764 39V18 4016 4044 4104 4161 4164
4256 4257 4264 4416 4564 4864 48C64 5063
5133 5143 51C64 51C65 51C66 51C67 51C68 51C69
52B13 52B23 52B33 5301 5305 5306 5330 5331
5340 5341 5349 5352 5353 5381 5513 5514
5516 5517 5564 5565 5600 5604 5605 5610
5623 5624 5625 6116 6147 6164 6167 6256
6257 6264 6267 6287 6301 6305 6306 6330
6331 6340 6341 6349 6352 6353 6380 6381
6665 7051 7052 7053 7056 7057 7058 7121E
7122E 7128E 7131 7132 7134 7137 7138 7142E

7602 7603 7610 7611 7620 7621 7640 7641
7642 7643 7649 7680 7681 7685 8128 8167
8168 82S23 8264 8266 8281 8416 8417 8418
8464 87C64 9044 9064 9114 9124 9128 9244
9864 99C68 18S030 18SA030 21256 25044 25045 25085
25088 25089 25168 25169 27010 27011 27128 27S180
27S181 27S185 27S190 27S191 27S200 27210 27256 27C256
27S280 27S281 27S290 27S291 27512 27513 27916 28S116
28S166 29611 29621 29625 29630 29631 29651 29661
29680 29681 37256 41256 48416 50256 50257 51S256
51C259 57256 74S188 74S287 74S288 74S387 74S472 74S474
74S475 74S570 74S571 74S572 74S573 76160 76161 76164
76165 76321 76641 81256 81257 81416 82S123 82S126
82S129 82S130 82S131 82S136 82S137 82S147 82S180 82S181

82S185 82S190 82S191 82S321 87S180 87S181 87S185 87S190
87S191 87S280 87S281 87S290 87S291 93415 93517 93425
93427 93436 93438 93446 93448 93Z450 93Z451 93452
93453 93Z510 93Z511 93512 99C641 63S1681

Directory of MOTO.LIB

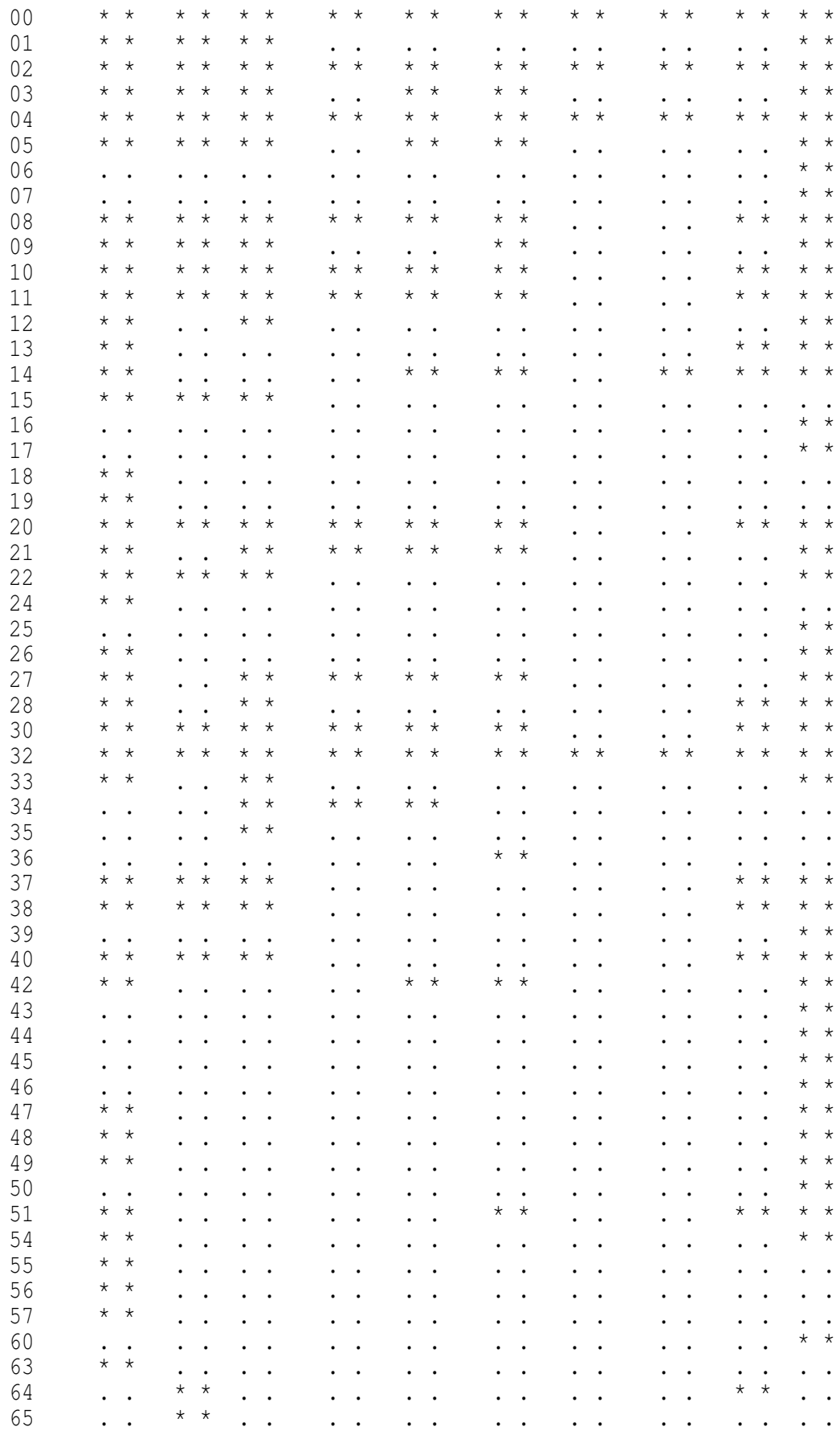
6800 68A00 68B00 6801 6801EM 6801NM 6802
6802NS 6803 6803EM 6808 6809 6809E 68HC09E
6810 6821 6822 6829 68H34 6835 6839
6840 6844 6845 6846 6847 6847Y 6850
6852 6854 6875 6875A 68000CC 68000D 68000GA
68008D 68008Q 68010CC 68010D 68010GA 6801U4 6801U4EM
680104NM 68020 6803U4 6803U4EM 6804J2 6804P2 68HC04P2
68HC04P3 6805K2 6805P2 6805R2 6805T2 6805U2 6805K3
6805R3 6805U3 6805P4 6805P6 68120EM 68120NM 68120SC
68121EM 68121NM 58121SC 68440 68451 68681 68901
146805E3

Directory of RF.LIB

AIR INDUCTPOR AIR INDUCTOR VARIABLE AIR INDUCTOR WIPER
AIR T INDUCTOR ANTENNA DIPOLE ANTENNA NETWORK
ANTENNA-1 ANTENNA-2 ARROW
BOX CAPACITOR FEED CIRCLE
COAX PLUG COAX RECEPTACLE CONNECTOR COAX
CONNECTOR COAX-F CONNECTOR COAX CRT
ENVELOPE FLYBACK GROUNDED CAP
HOTLINE JACK PENTODE PHASE SHIFTER
PHASE SHIFTER BETA PHASE SHIFTER THETA PSB
PST RF DPDT RF SPDT
STDA115 STDA116 STEPUP
SWITCH NETWORK TETRODE TRANSMISSION LINE
TRANSMISSION TRIODE TRIODE PCAP
.pn256

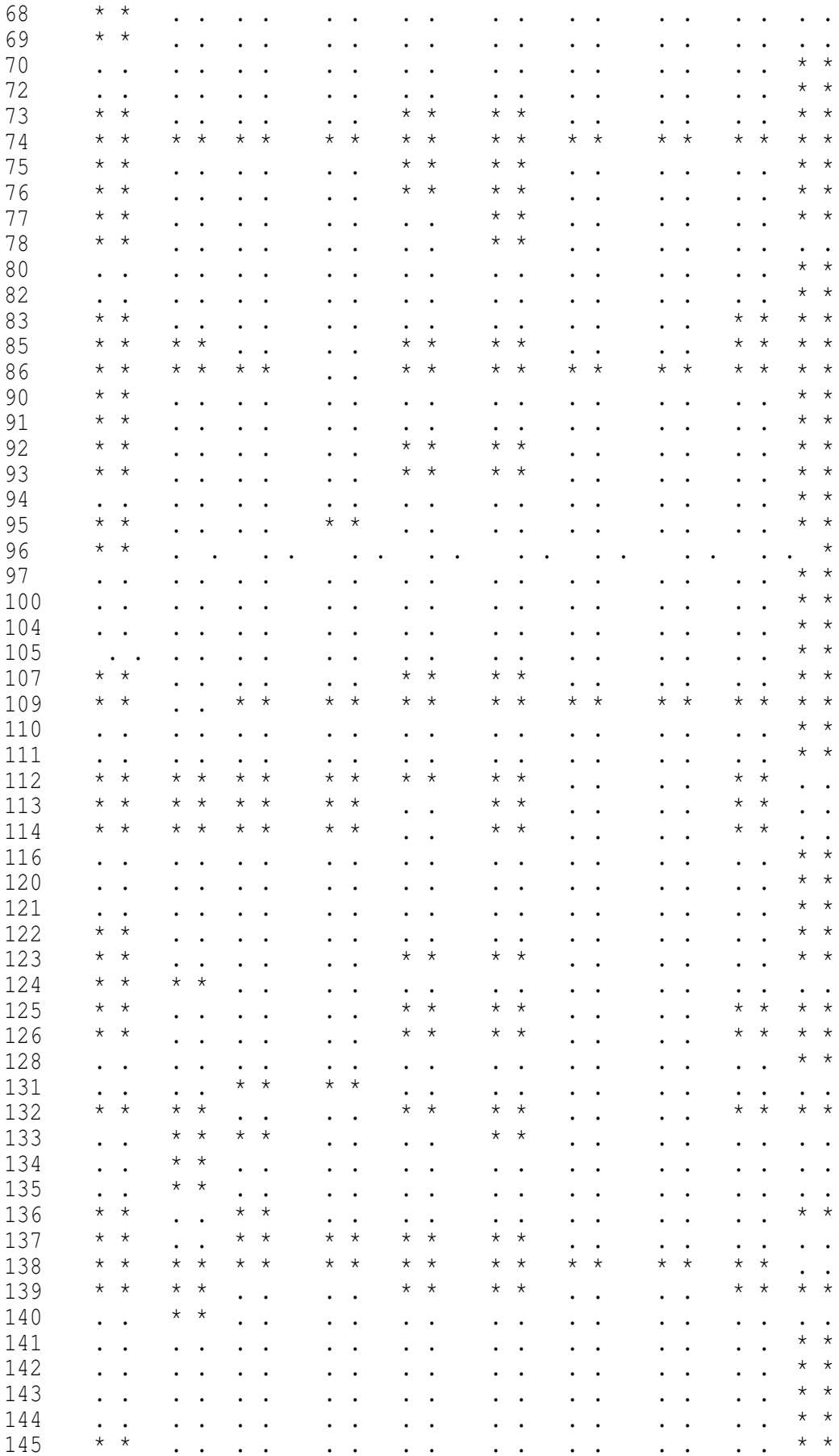
Directory of TTL.LIB

74LS 74S 74ALS 74AS 74HCT 74HC 74ACT 74AC 74F 74



Directory of TTL.LIB Continued

74LS 74S 74ALS 74AS 74HCT 74HC 74ACT 74AC 74F 74



Directory of TTL.LIB Continued

	74LS	74S	74ALS	74AS	74HCT	74HC	74ACT	74AC	74F	741
147	* *	. .	. .	. .	* *	* *	. .	. .	. .	* *
148	* *	. .	. .	. .	. .	* *	. .	. .	* *	* *
150	. .	. .	. .	* *	. .	. .	. .	. .	. .	* *
151	* *	* *	* *	* *	* *	* *	* *	* *	* *	* *
152	* *	. .	. .	. .	. .	* *	. .	. .	. .	* *
153	* *	* *	* *	* *	* *	* *	* *	* *	* *	* *
154	* *	. .	. .	. .	* *	* *	. .	. .	. .	* *
155	* *	. .	. .	. .	* *	* *	. .	. .	. .	* *
156	* *	. .	. .	. .	. .	. .	. .	. .	. .	* *
157	* *	* *	* *	* *	* *	* *	* *	* *	* *	* *
158	* *	* *	* *	* *	* *	* *	* *	* *	* *	. .
159	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
160	* *	. .	* *	* *	* *	* *	. .	. .	* *	. .
161	* *	. .	* *	* *	* *	* *	* *	* *	* *	. .
162	* *	* *	* *	* *	* *	* *	. .	. .	* *	. .
163	* *	* *	* *	* *	* *	* *	* *	* *	* *	. .
164	* *	. .	* *	. .	* *	* *	. .	. .	* *	* *
165	* *	. .	* *	. .	* *	* *	. .	. .	. .	* *
166	* *	. .	* *	. .	* *	* *	. .	. .	* *	* *
167	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
168	* *	* *	* *	* *	. .	. .	. .	. .	* *	. .
169	* *	* *	* *	* *	* *	* *	. .	. .	* *	. .
170	* *	. .	. .	. .	. .	. .	. .	. .	. .	* *
171	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
172	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
173	* *	. .	. .	. .	* *	* *	. .	. .	. .	* *
174	* *	* *	* *	* *	* *	* *	. .	. .	* *	* *
175	* *	* *	* *	* *	* *	* *	. .	. .	* *	* *
176	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
177	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
178	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
179	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
180	. .	. .	. .	. .	. .	* *	. .	. .	. .	* *
181	* *	* *	. .	* *	* *	* *	. .	. .	* *	* *
182	. .	* *	. .	. .	* *	* *	. .	. .	* *	* *
183	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
184	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
185	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
190	* *	. .	* *	. .	* *	* *	. .	. .	* *	* *
191	* *	. .	* *	. .	* *	* *	* *	* *	* *	* *
192	* *	. .	* *	. .	* *	* *	. .	. .	* *	* *
193	* *	. .	* *	. .	* *	* *	* *	* *	* *	* *
194	* *	* *	. .	. .	* *	* *	. .	. .	* *	* *
195	* *	* *	. .	. .	* *	* *	. .	. .	* *	* *
196	* *	* *	. .	. .	. .	. .	. .	. .	. .	* *
197	* *	* *	. .	. .	. .	. .	. .	. .	. .	. .
198	. .	. .	. .	. .	. .	. .	. .	. .	* *	. .
199	. .	. .	. .	. .	. .	. .	. .	. .	* *	. .
221	* *	. .	. .	. .	* *	* *	. .	. .	. .	. .
226	. .	* *	. .	. .	. .	. .	. .	. .	. .	. .
237	. .	. .	. .	. .	* *	* *	. .	. .	. .	. .
240	* *	* *	* *	* *	* *	* *	* *	* *	* *	. .
241	* *	* *	* *	* *	* *	* *	* *	* *	* *	. .
242	* *	. .	* *	* *	* *	* *	. .	. .	* *	. .
243	* *	. .	* *	* *	* *	* *	. .	. .	* *	. .
244	* *	* *	* *	* *	* *	* *	* *	* *	* *	. .
245	* *	. .	* *	. .	* *	* *	* *	* *	* *	. .

	74LS	74S	74ALS	74AS	74HCT	74HC	74ACT	74AC	74F	741
246	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
247	* *	. .	. .	. .	. .	. .	. .	. .	. .	* *
248	* *	. .	. .	. .	. .	. .	. .	. .	. .	* *
249	* *	. .	. .	. .	. .	. .	. .	. .	. .	* *
250	. .	. .	. .	* *	. .	. .	. .	. .	. .	. .
251	* *	* *	* *	* *	* *	* *	* *	* *	* *	* *
253	* *	. .	* *	* *	* *	* *	* *	* *	* *	. .
257	* *	* *	* *	* *	* *	* *	* *	* *	* *	. .
258	* *	* *	* *	* *	* *	* *	* *	* *	* *	. .
259	* *	. .	* *	. .	* *	* *	. .	. .	* *	* *
260	. .	* *	. .	. .	. .	. .	. .	. .	* *	. .
261	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
266	* *	. .	. .	. .	. .	* *	. .	. .	. .	. .
268	. .	* *	. .	. .	. .	. .	. .	. .	. .	. .
273	* *	. .	* *	. .	* *	* *	. .	. .	* *	* *
274	. .	* *	. .	. .	. .	. .	. .	. .	. .	. .
275	* *	* *	. .	. .	. .	. .	. .	. .	. .	. .
276	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
278	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
279	* *	. .	. .	. .	. .	. .	. .	. .	. .	* *
280	* *	* *	. .	. .	* *	* *	* *	* *	* *	. .
282	. .	. .	. .	* *	. .	. .	. .	. .	. .	. .
283	* *	* *	. .	. .	* *	. .	* *	* *	* *	* *
284	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
285	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
286	. .	. .	. .	* *	. .	. .	. .	. .	. .	. .
290	* *	. .	. .	. .	. .	. .	. .	. .	. .	* *
292	* *	. .	. .	. .	. .	* *	. .	. .	. .	. .
293	* *	. .	. .	. .	. .	. .	. .	. .	. .	* *
294	* *	. .	. .	. .	. .	* *	. .	. .	. .	. .
295	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
298	* *	. .	. .	* *	. .	* *	. .	. .	* *	* *
299	* *	* *	* *	. .	* *	* *	* *	* *	* *	. .
320	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
321	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
322	* *	. .	. .	. .	. .	* *	. .	. .	* *	. .
323	* *	. .	* *	. .	* *	* *	* *	* *	* *	. .
347	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
348	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
350	. .	* *	. .	. .	. .	. .	. .	. .	* *	. .
352	* *	. .	* *	* *	. .	* *	. .	. .	* *	. .
353	* *	. .	* *	* *	. .	* *	. .	. .	* *	. .
354	* *	. .	. .	. .	* *	* *	. .	. .	. .	. .
355	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
356	* *	. .	. .	. .	* *	* *	. .	. .	. .	. .
357	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
365	* *	. .	* *	. .	* *	* *	. .	. .	* *	* *
366	* *	. .	* *	. .	* *	* *	. .	. .	* *	* *
367	* *	. .	* *	. .	* *	* *	. .	. .	* *	* *
368	* *	. .	* *	. .	* *	* *	. .	. .	* *	* *
373	* *	* *	* *	* *	* *	* *	* *	* *	* *	. .
374	* *	* *	* *	* *	* *	* *	* *	* *	* *	. .
375	* *	. .	. .	. .	. .	* *	. .	. .	. .	. .
376	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
377	* *	. .	. .	. .	* *	* *	. .	. .	* *	. .
378	* *	. .	. .	. .	. .	* *	. .	. .	* *	. .
379	* *	. .	. .	. .	. .	* *	. .	. .	* *	. .

Directory of TTL.LIB Continued

	74LS	74S	74ALS	74AS	74HCT	74HC	74ACT	74AC	74F	741
381	* *	* *	. .	. .	. .	* *	. .	. .	* *	. .
382	* *	. .	. .	. .	. .	* *	. .	. .	* *	. .
384	* *	. .	. .	. .	. .	. .	. .	. .	* *	. .
385	* *	. .	. .	. .	. .	. .	. .	. .	* *	. .
386	. .	. .	. .	. .	. .	* *	. .	. .	. .	. .
390	* *	. .	. .	. .	* *	* *	. .	. .	. .	. .
393	* *	. .	. .	. .	* *	* *	. .	. .	. .	. .
395	* *	. .	. .	* *	. .	. .	. .	. .	* *	. .
396	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
397	* *	. .	. .	. .	. .	. .	. .	. .	* *	. .
399	* *	. .	. .	. .	. .	. .	. .	. .	* *	. .
412	. .	* *	. .	. .	. .	. .	. .	. .	* *	. .
422	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
423	* *	. .	. .	. .	* *	* *	. .	. .	. .	. .
425	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
426	. .	. .	. .	. .	. .	. .	. .	. .	. .	* *
428	. .	* *	. .	. .	. .	. .	. .	. .	. .	. .
432	. .	. .	. .	. .	. .	. .	. .	. .	* *	. .
436	. .	* *	. .	. .	. .	. .	. .	. .	. .	. .
437	. .	* *	. .	. .	. .	. .	. .	. .	. .	. .
440	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
441	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
442	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
443	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
444	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
445	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
446	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
447	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
448	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
449	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
465	* *	. .	* *	. .	. .	. .	* *	. .	. .	. .
466	* *	. .	* *	. .	. .	. .	* *	. .	. .	. .
467	* *	. .	* *	. .	. .	. .	* *	. .	. .	. .
468	* *	. .	* *	. .	. .	. .	* *	. .	. .	. .
484	. .	* *	. .	. .	. .	. .	. .	. .	. .	. .
485	. .	* *	. .	. .	. .	. .	. .	. .	. .	. .
490	* *	. .	. .	. .	. .	* *	. .	. .	. .	* *
518	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
519	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
520	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
521	. .	. .	* *	. .	* *	* *	. .	. .	* *	. .
522	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
526	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
527	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
528	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
533	. .	. .	* *	* *	* *	* *	* *	* *	* *	. .
534	. .	. .	* *	* *	* *	* *	* *	* *	* *	. .
538	. .	. .	* *	. .	. .	. .	. .	. .	* *	. .
539	. .	. .	* *	. .	. .	. .	. .	. .	* *	. .
540	* *	. .	* *	. .	* *	* *	* *	* *	* *	. .
541	* *	. .	* *	. .	* *	* *	* *	* *	* *	. .
560	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
561	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
563	. .	. .	* *	. .	* *	* *	. .	. .	* *	. .
564	. .	. .	* *	. .	* *	* *	. .	. .	* *	. .
568	. .	. .	* *	. .	. .	. .	. .	. .	* *	. .
569	. .	. .	* *	. .	. .	. .	. .	. .	* *	. .

Directory of TTL.LIB Continued

	74LS	74S	74ALS	74AS	74HCT	74HC	74ACT	74AC	74F	741
573	. .	. .	* *	* *	* *	* *	. .	. .	* *	. .
574	. .	. .	* *	* *	* *	* *	. .	. .	* *	. .
575	. .	. .	* *	* *	. .	. .	. .	. .	. .	. .
576	. .	. .	* *	* *	. .	. .	. .	. .	. .	. .
577	. .	. .	* *	* *	. .	. .	. .	. .	. .	. .
580	. .	. .	* *	* *	. .	. .	. .	. .	. .	. .
588	. .	. .	. .	. .	. .	. .	. .	. .	* *	. .
589	* *	. .	. .	. .	. .	* *	. .	. .	. .	. .
590	* *	. .	. .	. .	* *	* *	. .	. .	. .	. .
591	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
592	* *	. .	. .	. .	* *	* *	. .	. .	. .	. .
593	* *	. .	. .	. .	* *	* *	. .	. .	. .	. .
594	* *	. .	. .	. .	. .	* *	. .	. .	. .	. .
595	* *	. .	. .	. .	. .	* *	. .	. .	* *	. .
596	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
597	* *	. .	. .	. .	* *	* *	. .	. .	* *	. .
598	* *	. .	. .	. .	* *	* *	. .	. .	. .	. .
599	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
604	* *	. .	. .	. .	. .	* *	. .	. .	* *	. .
605	* *	. .	. .	. .	. .	. .	. .	. .	* *	. .
606	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
607	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
608	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
618	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
619	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
620	* *	. .	* *	* *	* *	* *	* *	. .	* *	. .
621	* *	. .	* *	* *	. .	. .	. .	. .	* *	. .
622	* *	. .	* *	* *	. .	. .	. .	. .	* *	. .
623	* *	. .	* *	* *	* *	* *	* *	. .	* *	. .
638	* *	. .	* *	* *	. .	. .	. .	. .	. .	. .
639	* *	. .	* *	* *	. .	. .	. .	. .	. .	. .
640	* *	. .	* *	* *	* *	* *	* *	. .	. .	. .
641	* *	. .	* *	* *	. .	. .	. .	. .	. .	. .
642	* *	. .	* *	* *	. .	. .	. .	. .	. .	. .
643	* *	. .	* *	* *	* *	* *	* *	. .	. .	. .
644	* *	. .	* *	* *	. .	. .	. .	. .	. .	. .
645	* *	. .	* *	* *	. .	* *	. .	. .	. .	. .
646	* *	. .	* *	* *	* *	* *	* *	* *	* *	. .
647	* *	. .	* *	* *	. .	. .	. .	. .	* *	. .
648	* *	. .	* *	* *	* *	* *	* *	* *	* *	. .
649	* *	. .	* *	* *	. .	. .	. .	. .	* *	. .
651	* *	. .	* *	* *	* *	* *	. .	. .	* *	. .
652	* *	. .	* *	* *	* *	* *	. .	. .	* *	. .
653	* *	. .	* *	* *	. .	. .	. .	. .	* *	. .
654	* *	. .	* *	* *	. .	. .	. .	. .	* *	. .
668	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
669	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
670	* *	. .	. .	. .	* *	* *	. .	. .	. .	. .
671	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
672	* *	. .	. .	. .	. .	. .	. .	. .	. .	. .
673	* *	. .	. .	. .	. .	* *	. .	. .	* *	. .
674	* *	. .	. .	. .	. .	* *	. .	. .	* *	. .
677	. .	. .	* *	. .	. .	* *	. .	. .	. .	. .
678	. .	. .	* *	. .	. .	* *	. .	. .	. .	. .
679	. .	. .	* *	. .	. .	* *	. .	. .	. .	. .
680	. .	. .	* *	. .	. .	* *	. .	. .	. .	. .

```
681    * *   . .  . .    . .   . .    . .   . .    . .   . .  . .

     Directory of TTL.LIB  Continued

       74LS  74S  74ALS  74AS  74HCT  74HC  74ACT  74AC  74F  741
682    * *   . .  . .    . .   . .    * *   . .    . .   . .  . .
683    * *   . .  . .    . .   . .    . .   . .    . .   . .  . .
684    * *   . .  . .    . .   . .    * *   . .    . .   . .  . .
685    * *   . .  . .    . .   . .    . .   . .    . .   . .  . .
686    * *   . .  . .    . .   . .    * *   . .    . .   . .  . .
687    * *   . .  . .    . .   . .    . .   . .    . .   . .  . .
688    * *   . .  * *    . .   * *    * *   . .    . .   . .  . .
689    * *   . .  * *    . .   . .    . .   . .    . .   . .  . .
690    * *   . .  . .    . .   . .    * *   . .    . .   . .  . .
691    * *   . .  . .    . .   . .    * *   . .    . .   . .  . .
692    * *   . .  . .    . .   . .    * *   . .    . .   . .  . .
693    * *   . .  . .    . .   . .    * *   . .    . .   . .  . .
696    * *   . .  . .    . .   . .    * *   . .    . .   . .  . .
697    * *   . .  . .    . .   . .    * *   . .    . .   . .  . .
698    * *   . .  . .    . .   . .    * *   . .    . .   . .  . .
699    * *   . .  . .    . .   . .    * *   . .    . .   . .  . .
742    . .   . .  * *    . .   . .    . .   . .    . .   . .  . .
743    . .   . .  * *    . .   . .    . .   . .    . .   . .  . .
746    . .   . .  * *    . .   . .    . .   . .    . .   . .  . .
747    . .   . .  * *    . .   . .    . .   . .    . .   . .  . .
756    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
757    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
758    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
759    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
760    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
800    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
802    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
804    . .   . .  * *    * *   . .    * *   . .    . .   * *  . .
805    . .   . .  * *    * *   . .    * *   . .    . .   * *  . .
808    . .   . .  * *    * *   . .    * *   . .    . .   . .  . .
821    . .   . .  . .    * *   . .    . .   . .    . .   * *  . .
822    . .   . .  . .    * *   . .    . .   . .    . .   * *  . .
823    . .   . .  . .    * *   . .    . .   . .    . .   * *  . .
824    . .   . .  . .    * *   . .    . .   . .    . .   * *  . .
825    . .   . .  . .    * *   . .    . .   . .    . .   * *  . .
826    . .   . .  . .    * *   . .    . .   . .    . .   * *  . .
832    . .   . .  * *    * *   . .    * *   . .    . .   . .  . .
841    . .   . .  . .    * *   . .    . .   . .    . .   * *  . .
842    . .   . .  . .    * *   . .    . .   . .    . .   * *  . .
843    . .   . .  . .    * *   . .    . .   . .    . .   * *  . .
844    . .   . .  . .    * *   . .    . .   . .    . .   * *  . .
845    . .   . .  . .    * *   . .    . .   . .    . .   * *  . .
846    . .   . .  . .    * *   . .    . .   . .    . .   * *  . .
850    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
851    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
852    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
856    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
857    . .   . .  * *    * *   . .    . .   . .    . .   . .  . .
866    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
867    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
869    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
873    . .   . .  * *    * *   . .    . .   . .    . .   . .  . .
874    . .   . .  * *    * *   . .    . .   . .    . .   . .  . .
876    . .   . .  * *    * *   . .    . .   . .    . .   . .  . .
877    . .   . .  . .    * *   . .    . .   . .    . .   . .  . .
```

878	. .	. .	* *	* *	. .	. .	. .	. .	. .	. .	. .
879	. .	. .	* *	* *	. .	. .	. .	. .	. .	. .	. .

Directory of TTL.LIB Continued

	74LS	74S	74ALS	74AS	74HCT	74HC	74ACT	74AC	74F	741
880	. .	. .	* *	* *	. .	. .	. .	. .	. .	. .
881	. .	. .	. .	* *	. .	* *	. .	. .	* *	. .
882	. .	. .	. .	* *	. .	* *	. .	. .	* *	. .
885	. .	. .	. .	* *	. .	. .	. .	. .	. .	. .
888	. .	. .	. .	* *	. .	. .	. .	. .	. .	. .
888G	. .	. .	. .	* *	. .	. .	. .	. .	. .	. .
890	. .	. .	. .	* *	. .	. .	. .	. .	. .	. .
890G	. .	. .	. .	* *	. .	. .	. .	. .	. .	. .
1000	. .	. .	* *	* *	. .	. .	. .	. .	. .	. .
1002	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1003	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1004	. .	. .	* *	* *	. .	. .	. .	. .	. .	. .
1005	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1008	. .	. .	* *	* *	. .	. .	. .	. .	. .	. .
1010	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1011	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1020	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1032	. .	. .	* *	* *	. .	. .	. .	. .	. .	. .
1034	. .	. .	* *	* *	. .	. .	. .	. .	. .	. .
1035	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1036	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1240	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1241	. .	. .	. .	. .	. .	. .	. .	. .	* *	. .
1242	. .	. .	* *	. .	. .	. .	. .	. .	* *	. .
1243	. .	. .	* *	. .	. .	. .	. .	. .	* *	. .
1244	. .	. .	* *	. .	. .	. .	. .	. .	* *	. .
1245	. .	. .	* *	. .	. .	. .	. .	. .	* *	. .
1620	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1621	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1622	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1623	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1638	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1639	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1640	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1641	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1642	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1643	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1644	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
1645	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
2620	. .	. .	. .	* *	. .	. .	. .	. .	. .	. .
2623	. .	. .	. .	* *	. .	. .	. .	. .	. .	. .
2640	. .	. .	. .	* *	. .	. .	. .	. .	. .	. .
2645	. .	. .	. .	* *	. .	. .	. .	. .	. .	. .
8003	. .	. .	* *	. .	. .	. .	. .	. .	. .	. .
11000	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11002	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11004	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11008	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11010	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11011	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11013	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11014	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11020	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11021	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .

11027	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .	
11030	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .	
11032	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .	

Directory of TTL.LIB Continued

	74LS	74S	74ALS	74AS	74HCT	74HC	74ACT	74AC	74F	741
11034	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11074	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11109	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11112	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11132	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11133	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11139	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11151	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11153	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11157	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11158	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11160	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11161	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11162	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11163	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11168	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11169	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11174	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11175	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11181	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11190	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11191	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11192	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11193	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11194	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11240	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11241	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11244	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11245	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11251	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11253	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11257	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11258	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11280	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11286	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11299	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11323	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11352	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11353	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11373	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11374	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11640	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11643	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .
11881	. .	. .	. .	. .	. .	. .	* *	* *	. .	. .

1488
1489
75188
75189
.PN266
.HE Appendix B Schematic Design Tools

A P P E N D I X

D R I V E R C O N F I G U R A T I O N S

This Appendix discusses the configuration of the supported graphics boards and Printer drivers.

G R A P H I C S B O A R D C O N F I G U R A T I O N S

OrCAD/SDT is designed to support many of the popular color and monochrome graphics boards that are available on the market. The host system can be configured with both a color and monochrome board residing in the system at the same. Outlined below are the jumper and switch settings required for the supported graphics boards.

T H E I B M C O L O R G R A P H I C S A D A P T E R

There are no jumper or switch settings to modify on the IMB Color Graphics Adapter.

OrCAD/SDT supports the following modes:

320 x 200 4 color [Color Display] Driver Name is: CGA4 .DRV
640 x 200 B & W [Color Display] Driver Name is: CGA2 .DRV

I B M E N H A N C E D G R A P H I C S A D A P T E R

OrCAD/SDT supports either the standard Enhanced Graphics Adapter with the Graphics Memory Expansion Card Option (128K RAM). Refer to the Installation Instructions when installing the IBM Enhanced Graphics Adapter board.

The standard Enhanced Graphics Adapter (64K RAM) supports the following modes:

320 x 200 16 color [Color Display] Driver Name is: EGA16C2.DRV
640 x 350 2 color [Mono Display] Driver Name is: EGA2.DRV
640 x 350 4 color [Enhanced Color] Driver Name is: EGA4E.DRV

The Enhanced Graphics Adapter with the Graphics Memory Expansion Card Option (128K RAM), supports the following modes:

320 x 200 16 color [Color Display] Driver Name is: EGA16C1.DRV
640 x 200 16 color [Color Display] Driver Name is: EGA16C2.DRV
640 x 350 2 color [Mono Display] Driver Name is: EGA2.DRV
640 x 350 16 color [Enhanced Color] Driver Name is: EGA16E.DRV
.pa

H E R C U L E S G R A P H I C S C A R D

There are no jumper or switch settings to modify on the Hercules Graphics Card.

OrCAD/SDT supports the following mode:

720 x 348 2 color [Monochrome Display] Driver Name is: HGC2.DRV

T E C M A R G R A P H I C S M A S T E R

OrCAD/SDT supports Tecmar color and monochrome graphic modes. Set the jumpers and switches to their normal positions as recommended by Tecmar. The following settings are necessary when operating the graphics board in the following modes:

Monochrome:

720 x 348 2 color [Mono Display] Driver Name is: TGM2N.DRV
720 x 696 2 color [Mono Display] Driver Name is: TGM2I.DRV

JPR1 - Position "A" is Not Jumpered
Position "B" is Jumpered
Position "C" is Jumpered
JPR7 - Jumpered
SW1 - Up

Color Graphics Mode:

640 x 200 16 color [Color Display] Driver Name is: TGM16N.DRV
640 x 400 16 color [Color Display] Driver Name is: TGM16I.DRV
720 x 200 4 color [Color Display] Driver Name is: TGM4N.DRV
720 x 400 4 color [Color Display] Driver Name is: TGM4I.DRV

JPR1 - Position "A" is Jumpered
Position "B" is Not Jumpered
Position "C" is Jumpered
JPR7 - Not Jumpered
SW1 - Down

T A N D Y 2 0 0 0

The driver name is: TANDY2K.DRV

N C R

The driver names are: NCR2.DRV

NCR4.DRV

V E C T R I X

The driver names are: VECTRXAT.DRV

VECTRXPC.DRV

F O U R C O L O R M O D E C O N F I G U R A T I O N S

When the four color mode of operation is selected on color graphic boards, the following colors may be selected when configuring the color table in OrCAD/SDT.

Color Graphics Adapter 320 x 200 4 color (Color Display)
Tecmar Graphics Master 720 x 200 4 color (Color Display)
Tecmar Graphics Master 720 x 400 4 color (Color Display)

Configuration:

Color Selected	Color Displayed
Blue	Green
Green	Green
Cyan	Green
Red	Red
Magenta	Red
Brown	Brown
Dark-Gray	Brown
Light-Gray	Brown
Light-Blue	Green
Light-Green	Green
Light-Cyan	Green
Light-Red	Red
Light-Magenta	Red
Yellow	Brown
White	Brown

When the four color mode of operation is selected on the IBM Enhanced Graphics Adapter the following colors may be selected when Configuring the Color Table in OrCAD/SDT.

Enhanced Graphics Adapter 640 x 350 4 color
(Enhanced Color Display) (64K on EGA)
.pa
Configuration:

Color Selected	Color Dislayed
Blue	Blue
Green	Blue
Cyan	Blue
Red	Red
Magenta	Red
Brown	White
Dark-Gray	White
Light-Gray	White
Light-Blue	Blue
Light-Green	Blue
Light-Cyan	Blue
Light-Red	Red
Light-Magenta	Red
Yellow	White
White	White

P R I T E R D R I V E R C O N F I G U R A T I O N S

To configure the printer drivers, there are two files that must be contained in the same sub-directory or floppy disk. They are: 1) PRINTER.DRV AND 2) the desired printer driver used for

your specific printer. Outlined below, is a list of the supported printers and their associated driver names:

Toshiba P1340, P1350, P351:

Printer Driver Name is: TOSHIBA.DRV
Printer Support Driver Name is: PRINTER.DRV

Epson FX, RX Series:

Printer Driver Name is: EPSON.DRV
Printer Support Driver Name is: PRINTER.drv

Epson MX Series:

Printer Driver Name is: EPSONMX.DRV
Printer Support Driver Name is: PRINTER.DRV

Epson LQ1000 Series and NEC PS Series (P560/P565):

Printer Driver Name is: LQ1000.DRV
Printer Support Driver Name is: PRINTER.DRV

.pa

.HE

I N D E X

A

Abandon edits 4-80
Adapter, graphics B-1
Again 4-7
ANNOTATE 1-3, 5-18, 6-7, 6-91, to 6-93, 8-37, 8-38
Annotation file 5-32, 6-7
Auto pan 4-84

B

BACKANNO 1-5, 6-7 to 6-15, 8-37, 8-41
Backup files 4-84
Battery backup 6-58
Bill of material (see PARTLIST)
Bitmaps 7-47
Block 4-9
  drag 4-10
  export 4-13
  fixup 4-10
  get 4-12
  import 4-13
  move 4-9
  save 4-12
Block symbols 7-10
Commands
  Again 4-7
  Blok 4-9
  Conditions 4-15
  Delete 4-17
  Edit 4-19
  Find 4-31
  Get 4-33
  Hardcopy 4-39
  Jump 4-43
  Library 4-47
  Macro 4-51
  Place 4-61
  Quit 4-77
  Repeat 4-81
  Set 4-83
  Tag 4-91
  Zoom 4-91
COMPOSER 1-3, 7-1 to 7-4, 8-25
Conditions 4-15
CONFIG.SYS 6-66
Configuration
  color table 2-23
  display driver 2-17
  driver prefix 2-17
hierarchy buffer size 2-22
initial macro 2-21

Bus entry 4-65
Buses
conecting to module ports 6-51
labeling 6-44
placing 4-63
splitting 6-48

C

CLEANUP 1-5, 6-19 to 6-22 8-37, 8-41
Color table 2-23
Connectors 6-64, 7-53
Conversion bitmaps 7-50
Convert 4-36, 7-50
Copy (BLOCK Save) 4-12
Cursor coordinates 4-86

D

DECOMP 1-3, 7-1, 7-3 to 7-5
Defining an area 4-1
Delete 4-17
block 4-18
object 4-17
undo 4-11
DeMorgan 1-1, 4-26
Display driver 2-17
DRAFT 1-1, 2-3, 2-4, 3-1 to 3-9 7-2, 8-1, 8-5, 8-6, 8-25
Dragging buses 4-85
Driver configuration B-1
Driver prefix 2-17

E

Edit 4-19
labels 4-19
parts 4-23
power objects 4-20
sheet symbols 4-21
title block 4-27
Enter sheet 4-77
ERCHECK 1-4, 5-18, 5-33, 6-23 8-38, 8-42
Error bell 4-85
Exiting DRAFT 3-8, 4-80
Exporting files 4-13

F

File format 6-1
macro 4-59
was/is 6-17
include 6-72
Find 4-31
8-37, 8-41
CLEANUP 6-2, 6-19, 8-37, 8-41
commands 3-3
invoking 2-15
library files 2-19
library prefix 2-19
macro bufer size 2-22
macro file 2-21
plotter driver 2-18
printer driver 2-18
sample dual floppy example 2-25
sample hard disk example 2-26
worksheet prefix 2-20

Four color mode B-4

G

Get 4-33
Going of grid 4-87
Graphics boards B-1
Gridarrays 7-16
Grid dots 4-86
Grid Parameters 8-9
Grid references 4-86, 8-9

H

Hardcopy 4-39
destination 4-39
file mode 4-40
make hardcopy 4-41
print mode 4-41
width of paper 4-42
Hard disk
configuration example 2-25
installation 2-3
using with OrCAD/SDT 8-1
Hierarchy 5-1
elements of 5-2
simple 3-12, 5-1, 5-10
complex 3-12, 5-2, 5-24
Hierarchy buffer size 2-22
Hotpoint 4-68

I

Identifiers
Importing files 4-13
Initialize 4-79
Initial macro 2-21
Installation
dual floppy systems 2-3
hard disk systems 2-9
Invoking
ANNOTATE 5-18, 6-2, 6-7, 8-37, 8-38
BACKANNO 6-2, 6-15
Library 4-47
browse 4-48
directory 4-47
object files 7-1

COMPOSER 7-4, 8-25
configuration file 2-15
DECOMP 7-4
DRAFT 3-1
ERCHECK 5-18, 6-2, 6-23, 8-38, 8-42
NETLIST 5-21, 6-2, 6-29 8-39
PARTLIST 5-23, 5-37, 6-2 6-67, 8-44
PLOTALL 6-2, 6-75, 8-46
PRINTALL 6-2, 6-87, 8-46
TREELIST 5-20, 6-2, 6-91, 8-38

two floppy-disk systems 3-1, 6-5
utilities 6-2

J
Jump 4-43
reference 4-43
tag 4-43
X-location 4-44
Y-location 4-44
Junction 4-64

K
Keywords 7-30

L
Labels 4-66, 6-44
applications 4-66, 6-44, 6-50
bus mumber 4-67, 6-45, 6-50
comment 4-67
correct positioning 4-68
internal 4-67, 6-44, 6-50
Leave sheet 4-78
usage 3-9
Move 4-9
Multiple parts in package 4-23

N
NETLIST 1-4, 5-21, 5-34, 6-29, 6-66, 8-42
Non-orthogonal 4-85

O
Orientatoin 4-25
devices 5-25
labels 5-25
power object 5-31
Outline symbol 4-34, 4-36

P
Part definition 7-8, 7-35
PARTLIST 1-4, 5-23, 5-37 6-67 to 6-74, 8-44
source files 7-1
using 8-10
Library files 2-19
Library prefix 2-19
Loading worksheets 3-6

M
Macro 4-51
capture 4-53
delete 4-58
file format 4-59
initialize 4-58
list 4-58
nesting 4-60
pausing 4-56
read 4-59
terminating between commands 4-55
valid macro keys 4-52
write 4-58
Macro buffer size 2-22
Macro file 2-21
Macro prompts 4-85
Menu organization 3-2
Mirror 4-27, 4-36, 4-36
Module ports 4-69, 6-46, 6-51, 6-57, 8-28
bidirectional 4-70,
connecting to buses 6-47
input 4-70
output 4-70
transferring power 4-69 5-31, 6-58
unspecified 4-69
Mouse
installation 2-14
left mouse button 3-9, 4-85, 8-46
Ploter driver 2-17
Ploting 6-75
HI 6-85
hints 6-84
HP 6-85
scaling 6-83
suppressing title block 6-86
troubleshooting 6-81
wiring diagrams 6-75,6-79
Power
connecting to module ports 6-57
handling 6-53
in a hierarchy 6-58
isolating 6-58
objects 4-20, 5-31, 8-31
placing objects 8-31
transferring 5-31,6-58
Prefix definition 7-6,7-

Part value 4-25
editing 4-25
moving location 4-25
Parts 4-47
Placing 4-35
Retrieving 4-33
Parts per package 4-23
Pen plotter table 2-21
Pin definition 7-39
Pin names 7-39
Pin numbers 4-86
Place 4-61
bus 4-63
entry 4-65
junction 4-64
label 4-66
module port 4-69
power 4-71
sheet 4-73
wire 4-61
PLOTALL 1-4, 3-8, 5-37, 6-87
moving location 4-24
Repeat 4-81
Repeat parameters 4-87
Replication 4-9
Root sheet 5-2
Rubberband 4-9, 4-10, 4-85

S

Saving worksheets 3-7
Set 4-83
auto pan 4-84
backup files 4-84
cursor coordinates 4-86
disable title block 4-86
dragging buses 4-85
error bell 4-85
going off grid 4-87
grid dos 4-86
grid references 4-96
left mouse button 4-85
non-orthogonal 4-85
pin numbers 4-86
repeat parameters 4-87
worksheet size 4-86
repeat parameters 4-87
worksheet size 4-86
Sheet symbols 3-14, 4-73, 5-2
Size of Worksheet 4-86
Suspend to DOS 4-79
Symbol boundary 4-34

T

Tag 4-91
Title block disable 4-86
Tokens 7-30
TREELIST 1-3, 2-6, 5-20, 5-33,
PRONTALL 1-4, 3-8, 5-37, 6-87, 8-46
Printer driver 2-18
configurations B-5
Printing 4-41, 5-37
batch mode 4-40

Q

Quit 4-77
abandon edits 4-80
enter sheet 4-77
initialize 4-79
leave sheet 4-78
suspend to DOS 4-79
update file 4-78
write to file 4-78

R

Reference designators 4-19, 4-24, 6-8
editing 4-23

U

UNDO 4-18
Update file 4-78
Updating files 3-8
Utilities
using 8-37

W

Which device? 4-24
Wires 1-2, 4-61
drawing 8-16
non-orthgonal 4-85
placing 8-16
Worksheet
editing 8-34
Worksheet file structures 3-9
flat file 3-10
hierarchy 3-12, 5-1
one sheet 3-14
Worksheet prefix 2-20
Write to file 4-78

Z

Zoom 4-91

6-91
Troubleshooting
configuration 2-28
general 1-7
plotter 6-81
setup 2-28

.PN274

Place a wire from pin 1 of the upper 74LS138 at Y=2.60 to pin 1 of the lower 74LS138.This wire is vertical along X=1.30.

Place a wire from pin 2 of the upper 74LS138 at Y=2.50 to pin 2 of the lower 74LS138.This wire is vertical along X=1.20.

Place a short vertical wire on pin 3 of the upper 74LS138.It ends at(1.50,2.80).

Place wires from pins 15,14,13,and 12 of the upper 74LS138 to X=3.70.

Place a wire from pin 3 of the first 74LS08 to (8.50,4.30).This wire is vertical along X=7.80.

Place a wire from pin 3 of the second74LS08 to(8.50,4.40).This wire is vertical along X=7.70.

Place a wire from pin 3 of the third 74LS08 to(8.50,4.40).This wire is vertical along X=7.70.

Place a wire from pin 3 of the fourth 74LS08 to(8.50,4.60).This wire is vertical along X=7.80.

Making Junctions

Wires that cross do not represent a conection.To make a conection you must place a junction.A junction is unnecessary if the two wires are placed end to end.For example,you need a junction where the wire from pin 4 (G2A) of the lower decoder(the lower 74LS138) connects to the wire from pin 1 of the NAND gate.

Action: Placing a JunctionFrom the main command menu,select PLACE.From the PLACE menu,highlight Junction and press the left button.Move the mouse to where you want the Junction.Press the left button twice.You can now place another junction by moving the mouse to another location and pressing the left button twice.

.pn275

When you no longer have junctions to place,press the right button.You need junctions where the output from the 74LS00 con nects to the inputs to the 74LS08s and where the adress lines going into the upper decoder connect to the wires going to the lower decoder.Place junctions at the following loca tions:(1.30,2.50),(1.20,2.60),
(1.50,4.70),(2.30,4.50),and(2.50,5.70).

So far in module 1,you've connected the parts with wires and placed junctions where needed.In Module 2.you'll add a bus and labels as well as some more parts.

8.3.6. Editing the Title Block

Title block is located in the lower righthand corner of the worksheet.You saw it if you zoomed out to look at the whole sheet earlier.To get the title block quickly,select the JUMP command from the main command menu.Select reference from the JUMP menu and then select A then 1.Alternatively,you can just move the mouse and let the screen pan until the title block comes into view.

Action: Editing the Title Block. From the main com
mand menu,select the EDIT command.Make
sure that the cursor is within the title
block.Then press the left button twice.A
menu indicating the fields in the block appears.Select each field in turn and type the appropriate informaton.
For example,to enter the Revision code
highlight Revision code and press the left
button.The prompt Revision code?,appears on the prompt line.Type in the revision code(limited to two digits) followed by <ENTER>.
Notice that the title block does not yet show what tou typed.
Now highlight the next field,Title of
sheet.Continue through the fields until you've entered all the information you want.You can leave fields blank.For exam
ple,you have the option of four adress lines,but Figure 8-1 doesn't show any in use.
When you've completed entering the infor
mation in the title block,press the right
button.Notice that the information now appears on the screen.You are back at the main command level.

.pa

8.3.7 Updating the Worksheet

You can choose either to update the file or write to a file.The difference is that update defaults to the currently loaded file.For example,you gave the name tutor.sch when you invoked DRAFT,and the Update File command defaults to that name.If you choose to write to a file,DRAFT requests the name of the file you want to write to.

Action: Updating the file. From the main command me
nu,select QUIT.Then select Update file.Press the right button to return to the main command level.

8.3.8 Making a Hardcopy

Action: Making a Hardcopy. From the main command menu select HARDCOPY.The HARDCOPY menu appears.Select Print Mode.Then select Scale.

Now check that your printer has power and is online.Then select Make Hardcopy.

Note that the Epson MX driver does not distinguish between Compress and Scale mode.It always prints in Scale mode regardless of the setting.

8.3.9 Exiting DRAFT

Action: Exiting DRAFT.From the main command menu,select QUIT.Then select Abandon Edits.The DOS prompt appears.

This is the end of Module 1.Module 2 completes the schematic begun in Module 1.Module 3 gives examples of using the OrCAD utilities.

8.4 Module 2: Constructing a Worksheet,Part 2

Summary:This module completes the drawing of the sample schematic.It entails creating and placing a custom part,defining a macro,adding input and output module ports,placing a bus,and placing power symbols.The placing of parts abd the connecting of wires is
not treated with as much detail as in Module 1.

Module 1 required one parts libary,TTL.LIB.Module 2 requires three libraries:TTL.LIB,DEVICE.LIB,and a custom library.Figure 8.2 shows the worksheet at the conclusion of Module 2.

Figure 8.2.The Worksheet after Performing Module 2

8.4.1 Creating the Custom Library

Module 2 requires the use of a time-delay chip that is not in the OrCAD-supplied libraries.You must create a custom library that contains the part definition.To do that,1)create a library source file. 2)run COMPOSER on that source file to create a library file,and 3)reconfigure DRAFT to use the custom library. Section 7,Libraries,contains a detailed description of how to create a custom library.

To create the source file,use a text editor that makes a

pure ASCII file with no embedded control characters.This is the same kind of text editor,you would use to create program source files.For example either Wordstar in the non-document mode or EDLIN satisfies this requirement.

Action: Creating a Custom Library. Use a text editor to create the following text file.Call the file tutor.src and place it in the library directory.Enter the following text in the file.

```
PREFIX
END
'time delay'
6  4  1
L1  1   IN    'IN'
B1  12  OUT   '25'
B3  10  OUT   '75'
B5  8   OUT   '125'
B0  7   PWR   'GND'
T0  14  PWR   'VCC'
```

Then invoke the OrCAD utility COMPOSER as follows.
The command line below assumes that COMPOSER.EXE is in the directory \orcad and that tutor.src and tutor.lib are in the directory\orcad\library.

```
C:\ORCAD>COMPOSER\ORCAD\LIBRARY\TUTOR.SRC\ORCAD\LIBRARY\TUTOR.LIB
```

Then invoke DRAFT with the \C option and add the library tutor.lib.Ensure that all three of the required libraries are configured.

8.4.2. Invoking DRAFT and Setting a Macro

Action: Invoking DRAFT. Invoke DRAFT and specify the file tutor.sch that you created in Module 1.Enter the following

```
C:\ORCAD>DRAFT TUTOR.SCH
```

Notice when DRAFT comes up, the grid dots are not visible: nor are the X Y coordinates shown in the upper right corner.To set the environment as in Module 1,you must once again issue the SET command.However, if the Module 1 environment is one you plan to use frequently,it's convenient to define a macro.The action below defines the function key <F1> as a macro that makes the grid dots visible and starts up the X Y display.

Action: Defining a Macro. Press the left button to dis play the main command menu.Select Macro;that is ,highlight Macro and press the left button.Then select Capture.The message,Capture macro?,appears on the prompt line.Press the <F1> function key.The characters F1 appear on the prompt line.Then press <ENTER>.The message <macro> appears on the prompt line.This begins the macro definition.

Press <s>.The SET menu appears.Then press <g>. The grid parameters menu appears.Press <v>,then <y>.You are now back at the main command level.

Press <s>.The SET menu reappears.Press <x>,then <y>.You are now at the main command level.Press <m> to end the macro defini tion.

Note that in Module 1 you defined a macro on F2 and ended the macro capture with CTRL-END,rather than <m>.If you end a macro capture at a place in the menu hierarchy other than at the main command level,you must use CTRL-END.At the main command level either CTRL-END or <m> work.The macro defined above would accept either CTRL-END or <m>.The macro in Module 1 required CTRL-END.

Macros are usable right after you define them.But to use macros in future sessions,you must save them to a file.

Action: Saving Macros. From the main command level,press the left button.This displays the main command menu.Select Macro. Then select Write.The prompt,Write all macros to?,appears.Type the name of what will be your macro file(for example,macro.mac) and follow it with an <ENTER>.

The macro file is a text file.You can view it with the DOS type command.The macro definition for <F1> looks as follows{F1}= sgvysxy{}.You could have created the macro file with a text editor,and it would work just as well.

If you want the macro file to be loaded automatically every time you invoke DRAFT,you must reconfigure DRAFT and specify the macro file.Loading the macro file automatically does not mean executing the contained macros.It means making them available for use.

You can load the macro file manually.To do that,select MACRO from the main command menu.Then select Read and give the filename of the macro file.If you give just a filename,DRAFT assumes the macro file is in the same directory as DRAFT.EXE. Alternatively, you can specify a complete pathname.

8.4.3 Placing Module Ports

All signals that go from one sheet to another should be labeled with module ports.Signals that remain internal to the worksheet should be labeled as Internal,Bus Members,or Comments.

The worksheet from Module 1 has a number of unlabeled inputs and otputs.The next Action describes how to place the input module ports.

Action Placing Module Ports.From the main command level press the left button.The main command menu appears.Select PLACE.Then select Module Port.The prompt line reads Module Port Name?.Type the name of the first module port.In this case,it's A18.Follow it with an <ENTER>.Another menu appears,identifying the type of module port.Select Input.The module port appears on the screen.Move it so that its right tip touches the input line to pin 1 on the first decoder.Press the left button twice to place the module port.Once again the prompt line reads,Module Port Name?.Continuue the above procedure and place all the input

module ports.They are as follows.

Input	Module Port	Name	XY Location
A17			(0.60,2.60)
DACK0BRD\			(0.10,3.00)
			(0.50,3.20)
RAS			(0.60,4.50)
RAM ADDR SEL			(0.20,4.70)
DACK 0			(0.40,5.70)

This concludes the input module ports.Placing an output module port is similar.Just select Output instead of Input after you type the module port name.

Output	Module Port	Name	XY Location
RAS0\			(8.50,4.30)
RAS1\			(8.50,4.40)
RAS2\			(8.50,4.50)
RAS3\			(8.50,4.50)

Press the right button and return to the main command level.

8.4.4 Placing a Bus.

Placing a bus is similar to placing a wire.A bus appears on the screen as a thick wire.The column adress strobe lines from the top decoder are drawn as a bus.When drawing buses,yoy must pay special care when labeling buses that go off sheet.A bus label and a module port that is connected to a bus must be in the following format.

name[0..n]

where name is the name of the bus and n is the last number of the highest bus member.Both the bus label and bus module port must be in this format,but they don't have to have the same name.The part of the name outside the square brackets is called the bus prefix. The names ofindividual bus members consist of the prefix followed by the appropriate numerical suffix.The bus member prefix must be the same as the bus label prefix,but is not necessarily the same as the module port prefix.These requirements are illustrated in the following example.

Bus Label	Bus Module Port	Bus Members
CAS\[0..3]	CAS\[0..3]	CAS\0,CAS\1,etc.
BUSA[0..3]	CAS\[0..3]	BUSA0,BUSA1,etc.

Action: Placing a Bus. From the main command menu,select Place.Then select bus.Put the cursor where you want the bus to begin (3.80,2.60) and press the left button twice.Move the cursor to 8.10,2.90.Press the left button,highlight END, and press the left button again.Next place the output module port,CAS\[0..3].

Action: Placing Bus Entries. From the main command menu, select PLACE. Then select Bus (entry).A slash appears at the cursor tip.For this particular schematic,you want a backslash rather than a slash.(This decision is cosmetic only.)Press the left button,highlight the \ and press the left button again.Move the cursor so that the backslash connects the wire from pin 15 to the top decoder to the top of the bus .press the left button twice.Connect the other three bus members similary.Press the right button to return to the main command level.

Action: Placing Bus Member Labels. From the main command menu,select PLACE. Then select Label.The prompt,Label?,appears on the prompt line.Type the label,CAS\0,followed by an <ENTER>.Another menu appears.Select Bus Member.Move the bus member label until it rests on the wire output from pin 15 of the top decoder. This is XY location (3.70,2.50).Press the left button twice.The prompt Label?,reappears.Place the remaining three bus member labels .Then press the right button to return to the main command level.

Action: Placing a Bus Label.From the main command menu,select PLACE.Then select Label.The prompt,Label?,appears on the prompt line.Type the label,CAS\[0..3],followed by an <ENTER>.Another menu appears.Select Internal.Move the cursor and place the bus label on the bus.A good location is (4.30,2.90).

8.4.5 Placind Power Objects

The lower decoder takes +5V on pin 3;the upper decoder has its pin 3 grounded.Both the ground symbol and the +5V symbol are power objects,even tough the ground symbol comes from a library (DEVICE.LIB) and the +5V comes from the PLACE menu.

Action: Placing Power Objects.From the main command menu, select PLACE.Then select POWER.A list of commands appears on the prompt line.
The power symbol appears on the screen,and it moves with the

cursor.The default value is VCC.To change the value,press the left button and select Value.Backspace over the VCC and type +5V followed by an <ENTER>.Then press the left button and select Orientation.Then select Bottom.Now move the cursor to pin 3 on the lower decoder and press the left button twice.The other power symbol is a ground symbol belonging to DEVICE.LIB.It is called GND POWER.Obtain this part in the same way you obtained the decoders and gates in Module 1.Place it at pin 3 of the upper decoder.

Note that the devices placed on the worksheet do not show any power pins.However,the part definitions do define power pins, and power pins will appear in the netlist (see Module 3).The invisible power pins are labeled VCC and GND.

To make a netlist connection to the VCC pins,you must place a power object whose value is VCC in the worksheet.If you place another power object with value +5V,the NETLIST utility will assume two separate power supplies,one called +5V and one called VCC.If you want those to be the same power supply you must con nect the two power objects together .

Action: Connecting Power Objects.Place another power ob ject with value VCC at (7.40,0.70).Also,place a power object with value +5V at (7.90,0.70).Connect them to each other with a wire.

8.4.6 A Complete Circuit

Now is a good time to update the work file with the pdate command in the QUIT menu.Figure 8.3 shows how your worksheet looks so far.You can print it out with the HARDCOPY command.

Figure 8.3 Current State of the Worksheet

8.4.7 Placing More Parts

What remains to be done in Module 2 is to add a portion of the circuit that delivers the RAS and CAS\signals.This derive from the XMEMW\ and XMEMR\ signals (not yet shown).This involves placing three more parts on the worksheet.

Action: Placing Three More Parts. Place a 74LS00 from TTL.LIB at (6.50,1.60).Place its DeMorgan equivalent at 3.00,0.90.Then,place the custom part Time Delay from TUTOR.LIB at (4.80,1.00).

Action: Drawing More Wires. Draw the two wires input for the leftmost 74LS00.The wires begin at (1.00,1.00) and (1.00,1.20).Connect the output of the leftmost 74LS00 to

pin 1 on the time delay chip.Then,connect pins 8 and 12 to the inputs on the second 74LS00.Finally,bring a wire from pin 10 of the time delay chip to the right end of the worksheet, vertically even with the end of the bus.

Action: Placing Module Ports. Place three module ports .XMEMW\ and XMEMR\ are input to the first 74LS00,and ADDR DEL is output from the time delay chip.

8.4.8 Editing the Worksheet

First of all you must delete the RAS and CAS\ lines going into the decoders.These signals are no longer input module ports.

Action: Deleting the RAS Line. Put the cursor on the RAS line,around (1.80,4.50).From the main command menu select DELETE.Then from the DELETE menu select Object.Now press the left button twice.The wire is deleted up to the junction.Put the cursor on the RAS input module port and press the left button twice.The module port is now deleted.

The signal RAS comes from the output of the first 74LS00 and goes to pin 6 of the lower decoder.You need to place a junction between the first 74LS00 and the time delay chip.

Action Placing a Junction. Move the cursor to the wire that goes from the output of the first 74LS00 to the input of the time delay chip.(4.10,1.10) is a reasonable location.Place a junction.Then draw a wire down to (4.10,3.60).Then,turn left and draw a wire to (2.30,3.60).This lines up with the junction remai ning from the RAS input you deleted earlier.Draw a wire down to that junction.This is the RAS line.

So far in this tutorial,Left Button has been set to NO.If you've performed the designated Actions,you now have a pretty good feeling about how the mouse buttons operate.Try setting Left Button to YES,and observe that the user interface to the commands is somewhat different.

Action: Setting Left Button to YES. From the main command menu,select SET.Then,select Left Button,followed by Yes.Now delete the CAS\ input to the upper decoder.The new CAS\ line will come from the output of the second 74LS00.

Action: Deleting the CAS\Line. Move the cursor to the CAS\ input of the top decoder at (1.10,3.20).From the main command menu,select DELETE.Then,highlight Object and press the left button twice.The wire is now deleted.Put the cursor over the CAS\ module port.Press the left button.The module port is now de leted.From the main command menu ,select PLACE,then Wire.Place the cursor at the pin 5 of the second decoder at (1.50,3.20).Press the left button and begin drawing in the left

direction until (1.10,3.20).Then press the left button and go up until (1.10,2.20).Press the left button and go right until (7.40,2.20).Press the left button and go up to the output of the NAND gate (7.40,1.80).Press the left button and hold it down.Move the highlight to End and raise the left button.

Notice that drawing lines with Left Button set to YES re duces the number of button presses you must perform.It associates one of the button presses with a button release.

The schematic is now complete ,except for two labels.The final step is to label the RAS and CAS\ lines.

Action: Placing More Labels. From the main command menu ,select PLACE,then Label.At the prompt Label?,type CAS\ followed by an <ENTER>.Select Internal.Move the label to (7.40,2.00) and press the left button .The prompt Label? appears again.Type RAS,followed by an <ENTER>.Select Internal.Move the label to (4.10,1.90) and press the left button.Then press the right button to return to the main command level.

Ensure to update your work file.The Update command is under the QUIT menu.You may also to make a hardcopy of your design.Then, exit DRAFT by choosing Abandon edits from the QUIT menu.

Action: Exiting DRAFT. Update the work file.If you want a hardcopy,make one.Then,exit DRAFT.

8.5 Module 3: Using the OrCAD Utilities

Summary: This module shows how to run all the appropriate OrCAD utilities on the work file created in the two previous modules.

The work file constructed in Modules 1 and 2 is a one sheet schematic .Those utilities that deal with hierarchical structures or flat file structures are discussed only briefly in this mo dule.When you invoke one of the OrCAD utilities on a one sheet schematic ,append the switch \o to the command line.

The OrCAD utilities are as follows.

ANNOTATE This utility scans an input file and automatically updates reference designators.This includes updating the corres ponding pin numbers that are associated with a particular instan ce of a device with multiple parts per package.ANNOTATE modifies your work file;but it creates a backup file that contains the

orginal copy of the work file.

BACKANNO Use this utility when you want to update the reference designators after you've run ANNOTATE.For example,as sume that you sent your design out for layout;and when it comes back,what you has designated as U1A is now U1B.You can run BACKANO and supply as input your initial work file and a was/is text file that contains the new translation.

CLEANUP This utility scans a work file and checks for overlapping parts.It removes duplicate or overlapping wires,buses,and junctions.It displays warning messages advising you of duplicate objects.CLEANUP may modify your work file;but it creates a backup file that contains the original copy of your work file.CLEANUP does not check for objects overlapping buses (including wire entries of a bus overlapping bus entries to a bus).

ERRCHECK This utility performs a clasical electrical rules check.It flags unused inputs on parts,unlabeled wires connected to a bus,and invalid connections,such as two outputs wired toghe ther.

NETLIST This utility generates a net and wire list in a number of possible formats.The default format is EDIF.

PARTLIST This utility creates a list of the parts in your work file.If you have additional information that you want in cluded in that list,you can construct an Include file.

PRINTALL,PLOTALL These utilities are not discussed in this tutorial.They are used for printing and plotting schematics.

TREELIST This utility is not discussed in this tutorial.It scans a hierarchical structure and displays the sheet names,sheet pathnames,and optionally the date of last modification.

8.5.1 ANNOTATE

ANNOTATE is probably the first utility you should run .Other utilities report information about your work file, and ,if you run ANNOTATE first,you ensure that the informationis reported in terms of the updated reference designators.With ANNOTATE you can choose to have the updated information merged with the workfile or placed in an annotation file.The purpopse of an annotation file is to store the reference designators of a complex hierar chy.This example in this tutorial is a one sheet schematic and has no need of an annotation file.To run ANNOTATE on the workfile

tutor.sch type the following.

```
C:\ORCAD>ANNOTATE TUTOR.SCH/M/O
```

This example assumes that tutor.sch is in the sheet directo ry.The /m tells annotate to merge the updated information into the work file.If you left the /m out,ANNOTATE would abort.Without a/m,ANNOTATE expects two filenames.The first is the source file;the second is the name of what will be annotation file.Figure 8.4 shows how the worksheet looks after running ANNOTATE.Notice the updated reference designators on the devices with multiple parts per package.For example the U?A on the 74LS00s connected to the output of the lower decoder changed to U1A,U1B,U1C,and U1D.THey are all parts of the same package,and their pin numbers changed accordingly.

Figure 8.4 The Worksheet after Running ANNOTATE